Numerical Solution of

Partial Differential Equations

by

Gordon C. Everstine

21 January 2010

This book was typeset with LaTeX 2_ε (MiKTeX).

Preface

These lecture notes are intended to supplement a one-semester graduate-level engineering course at The George Washington University in numerical methods for the solution of partial differential equations. Both finite difference and finite element methods are included. The main prerequisite is a standard undergraduate calculus sequence including ordinary differential equations. In general, the mix of topics and level of presentation are aimed at upper-level undergraduates and first-year graduate students in mechanical, aerospace, and civil engineering.

Gordon Everstine
Gaithersburg, Maryland
January 2010

Contents

List of Figures

1 Numerical Solution of Ordinary Differential Equations

An *ordinary differential equation* (ODE) is an equation that involves an unknown function (the dependent variable) and some of its derivatives with respect to a single independent variable. An nth-order equation has the highest order derivative of order n:

$$f\left(x, y, y', y'', \ldots, y^{(n)}\right) = 0 \quad \text{for} \quad a \le x \le b, \tag{1.1}$$

where $y = y(x)$, and $y^{(n)}$ denotes the nth derivative with respect to x. An nth-order ODE requires the specification of n conditions to assure uniqueness of the solution. If all conditions are imposed at $x = a$, the conditions are called *initial conditions* (I.C.), and the problem is an *initial value problem* (IVP). If the conditions are imposed at both $x = a$ and $x = b$, the conditions are called *boundary conditions* (B.C.), and the problem is a *boundary value problem* (BVP).

For example, consider the initial value problem

$$\begin{cases} m\ddot{u} + ku = f(t) \\ u(0) = 5, \ \dot{u}(0) = 0, \end{cases} \tag{1.2}$$

where $u = u(t)$, and dots denote differentiation with respect to the time t. This equation describes a one-degree-of-freedom mass-spring system which is released from rest and subjected to a time-dependent force, as illustrated in Fig. 1. Initial value problems generally arise in time-dependent situations.

An example of a boundary value problem is shown in Fig. 2, for which the differential equation is

$$\begin{cases} EIu''(x) = M(x) = \dfrac{FL}{2}x - Fx\left(\dfrac{x}{2}\right) \\ u(0) = u(L) = 0, \end{cases} \tag{1.3}$$

where the independent variable x is the distance from the left end, u is the transverse displacement, and $M(x)$ is the internal bending moment at x. Boundary value problems generally arise in static (time-independent) situations. As we will see, IVPs and BVPs must be treated differently numerically.

A system of n first-order ODEs has the form

$$\begin{cases} y_1'(x) = f_1(x, y_1, y_2, \ldots, y_n) \\ y_2'(x) = f_2(x, y_1, y_2, \ldots, y_n) \\ \qquad\qquad \vdots \\ y_n'(x) = f_n(x, y_1, y_2, \ldots, y_n) \end{cases} \tag{1.4}$$

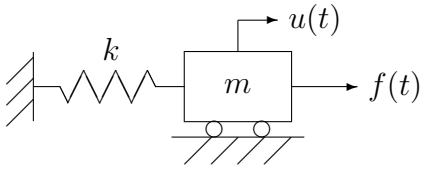


Figure 1: 1-DOF Mass-Spring System.

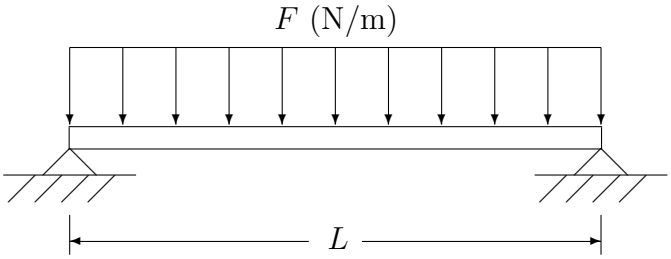


Figure 2: Simply-Supported Beam With Distributed Load.

for $a \leq x \leq b$. A single nth-order ODE is equivalent to a system of n first-order ODEs. This equivalence can be seen by defining a new set of unknowns $y_1, y_2, \ldots, y_n$ such that $y_1 = y$, $y_2 = y'$, $y_3 = y''$, $\ldots$, $y_n = y^{(n-1)}$. For example, consider the third-order IVP

$$\begin{array}{l} y''' = xy' + e^x y + x^2 + 1, \quad x \geq 0 \\ y(0) = 1,\ y'(0) = 0,\ y''(0) = -1. \end{array} \tag{1.5}$$

To obtain an equivalent first-order system, define $y_1 = y$, $y_2 = y'$, $y_3 = y''$ to obtain

$$\left\{ \begin{array}{l} y_1' = y_2 \\ y_2' = y_3 \\ y_3' = xy_2 + e^x y_1 + x^2 + 1 \end{array} \right. \tag{1.6}$$

with initial conditions $y_1(0) = 1$, $y_2(0) = 0$, $y_3(0) = -1$.

1.1 Euler's Method

This method is the simplest of the numerical methods for solving initial value problems. Consider the IVP

$$\left\{ \begin{array}{l} y'(x) = f(x, y), \quad x \geq a \\ y(a) = \eta. \end{array} \right. \tag{1.7}$$

To effect a numerical solution, we discretize the x-axis:

$$a = x_0 < x_1 < x_2 < \cdots,$$

where, for uniform spacing,

$$x_i - x_{i-1} = h, \tag{1.8}$$

and h is considered small. With this discretization, we can approximate the derivative $y'(x)$ with the forward finite difference

$$y'(x) \approx \frac{y(x+h) - y(x)}{h}. \tag{1.9}$$

If we let y_k represent the numerical approximation to $y(x_k)$, then

$$y'(x_k) \approx \frac{y_{k+1} - y_k}{h}. \tag{1.10}$$

Thus, a numerical (difference) approximation to the ODE, Eq. 1.7, is

$$\frac{y_{k+1} - y_k}{h} = f(x_k, y_k), \quad k = 0, 1, 2, \ldots \tag{1.11}$$

or

$$\begin{cases} y_{k+1} = y_k + hf(x_k, y_k), & k = 0, 1, 2, \ldots \\ y_0 = \eta. \end{cases} \tag{1.12}$$

This recursive algorithm is called *Euler's method.*

1.2 Truncation Error for Euler's Method

There are two types of error that arise in numerical methods: truncation error (which arises primarily from a discretization process) and rounding error (which arises from the finiteness of number representations in the computer). Refining a mesh to reduce the truncation error often causes the rounding error to increase.

To estimate the truncation error for Euler's method, we first recall Taylor's theorem with remainder, which states that a function $f(x)$ can be expanded in a series about the point $x = c$:

$$f(x) = f(c) + f'(c)(x-c) + \frac{f''(c)}{2!}(x-c)^2 + \cdots + \frac{f^{(n)}(c)}{n!}(x-c)^n + \frac{f^{(n+1)}(\xi)}{(n+1)!}(x-c)^{n+1}, \tag{1.13}$$

where ξ is between x and c. The last term in Eq. 1.13 is referred to as the *remainder* term. Note also that Eq. 1.13 is an equality, not an approximation.

In Eq. 1.13, let $x = x_{k+1}$ and $c = x_k$, in which case

$$y(x_{k+1}) = y(x_k) + hy'(x_k) + \frac{1}{2}h^2 y''(\xi_k), \tag{1.14}$$

where $x_k \leq \xi_k \leq x_{k+1}$.

Since y satisfies the ODE, Eq. 1.7,

$$y'(x_k) = f(x_k, y(x_k)), \tag{1.15}$$

where $y(x_k)$ is the actual solution at x_k. Hence,

$$y(x_{k+1}) = y(x_k) + hf(x_k, y(x_k)) + \frac{1}{2}h^2 y''(\xi_k). \tag{1.16}$$

Like Eq. 1.13, this equation is an equality, not an approximation.

By comparing this last equation to Euler's approximation, Eq. 1.12, it is clear that Euler's method is obtained by omitting the remainder term $\frac{1}{2}h^2 y''(\xi_k)$ in the Taylor expansion of $y(x_{k+1})$ at the point x_k. The omitted term accounts for the truncation error in Euler's method *at each step.* This error is a *local* error, since the error occurs at each step regardless of the error in the previous step. The accumulation of local errors is referred to as the *global* error, which is the more important error but much more difficult to compute.

Most algorithms for solving ODEs are derived by expanding the solution function in a Taylor series and then omitting certain terms.

1.3 Runge-Kutta Methods

Euler's method is a first-order method, since it was obtained by omitting terms in the Taylor series expansion containing powers of h greater than one. To derive a second-order method, we again use Taylor's theorem with remainder to obtain

$$y(x_{k+1}) = y(x_k) + hy'(x_k) + \frac{1}{2}h^2y''(x_k) + \frac{1}{6}h^3y'''(\xi_k) \tag{1.17}$$

for some ξ_k such that $x_k \leq \xi_k \leq x_{k+1}$. Since, from the ODE (Eq. 1.7),

$$y'(x_k) = f(x_k, y(x_k)), \tag{1.18}$$

we can approximate

$$y''(x) = \frac{df(x, y(x))}{dx} = \frac{f(x+h, y(x+h)) - f(x, y(x))}{h} + O(h) \tag{1.19}$$

where we use the "big O" notation $O(h)$ to represent terms of order h as $h \to 0$. [For example, $2h^3 = O(h^3)$, $3h^2 + 5h^4 = O(h^2)$, $h^2O(h) = O(h^3)$, and $-287h^4e^{-h} = O(h^4)$.] From these last two equations, Eq. 1.17 can then be written as

$$y(x_{k+1}) = y(x_k) + hf(x_k, y(x_k)) + \frac{h}{2}[f(x_{k+1}, y(x_{k+1})) - f(x_k, y(x_k))] + O(h^3), \tag{1.20}$$

which leads (after combining terms) to the difference equation

$$y_{k+1} = y_k + \frac{1}{2}h[f(x_k, y_k) + f(x_{k+1}, y_{k+1})]. \tag{1.21}$$

This formula is a second-order approximation to the original differential equation $y'(x) = f(x, y)$ (Eq. 1.7), but it is an inconvenient approximation, since y_{k+1} appears on *both* sides of the formula. (Such a formula is called an *implicit* method, since y_{k+1} is defined implicitly. An *explicit* method would have y_{k+1} appear only on the left-hand side.)

To obtain instead an explicit formula, we use the approximation

$$y_{k+1} = y_k + hf(x_k, y_k) \tag{1.22}$$

to obtain

$$y_{k+1} = y_k + \frac{1}{2}h[f(x_k, y_k) + f(x_{k+1}, y_k + hf(x_k, y_k))]. \tag{1.23}$$

This formula is the Runge-Kutta formula of second order. Other higher-order formulas can be derived similarly. For example, a fourth-order formula turns out to be popular in applications.

We illustrate the implementation of the second-order Runge-Kutta formula, Eq. 1.23, with the following algorithm. We first make the following three definitions:

$$a_k = f(x_k, y_k), \tag{1.24}$$

$$b_k = y_k + ha_k, \tag{1.25}$$

$$c_k = f(x_{k+1}, b_k), \tag{1.26}$$

in which case

$$y_{k+1} = y_k + \frac{1}{2}h(a_k + c_k). \tag{1.27}$$

The calculations can then be performed conveniently with the following spreadsheet:

k	x_k	y_k	a_k	b_k	c_k	y_{k+1}
0						
1						
2						
$\vdots$						

1.4 Systems of Equations

The methods just derived can be extended directly to systems of equations. Consider the initial value problem involving two equations:

$$\begin{cases} y'(x) = f(x, y(x), z(x)) \\ z'(x) = g(x, y(x), z(x)) \\ y(a) = \eta, \quad z(a) = \xi. \end{cases} \tag{1.28}$$

We recall from Eq. 1.12 that, for one equation, Euler's method uses the recursive formula

$$y_{k+1} = y_k + hf(x_k, y_k). \tag{1.29}$$

This formula is directly extendible to two equations as

$$\begin{cases} y_{k+1} = y_k + hf(x_k, y_k, z_k) \\ z_{k+1} = z_k + hg(x_k, y_k, z_k) \\ y_0 = \eta, \quad z_0 = \xi. \end{cases} \tag{1.30}$$

We recall from Eq. 1.23 that, for one equation, the second-order Runge-Kutta method uses the recursive formula

$$y_{k+1} = y_k + \frac{1}{2}h[f(x_k, y_k) + f(x_{k+1}, y_k + hf(x_k, y_k))]. \tag{1.31}$$

For two equations, this formula becomes

$$\begin{cases} y_{k+1} = y_k + \frac{1}{2}h[f(x_k, y_k, z_k) + f(x_{k+1}, y_k + hf(x_k, y_k, z_k), z_k + hg(x_k, y_k, z_k))] \\ z_{k+1} = z_k + \frac{1}{2}h[g(x_k, y_k, z_k) + g(x_{k+1}, y_k + hf(x_k, y_k, z_k), z_k + hg(x_k, y_k, z_k))] \\ y_0 = \eta, \quad z_0 = \xi. \end{cases} \tag{1.32}$$

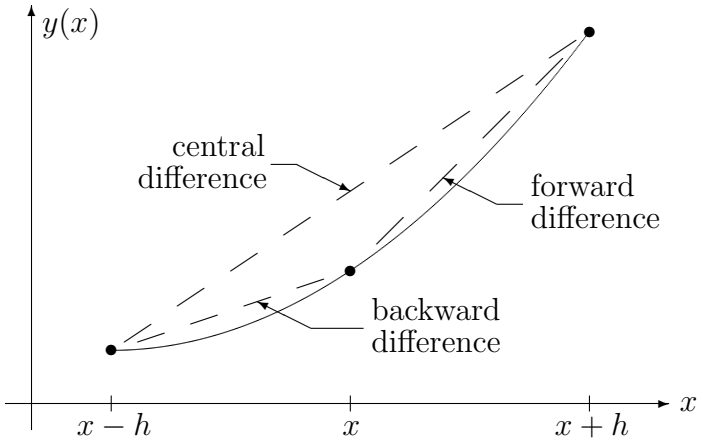


Figure 3: Finite Difference Approximations to Derivatives.

1.5 Finite Differences

Before addressing boundary value problems, we want to develop further the notion of finite difference approximation of derivatives.

Consider a function $y(x)$ for which we want to compute the derivative $y'(x)$ at some point x. If we discretize the x-axis with uniform spacing h, we could approximate the derivative using the *forward difference* formula

$$y'(x) \approx \frac{y(x+h) - y(x)}{h}, \tag{1.33}$$

which is the slope of the line to the right of x (Fig. 3). We could also approximate the derivative using the *backward difference* formula

$$y'(x) \approx \frac{y(x) - y(x-h)}{h}, \tag{1.34}$$

which is the slope of the line to the left of x. Since, in general, there is no basis for choosing one of these approximations over the other, an intuitively more appealing approximation results from the average of these formulas:

$$y'(x) \approx \frac{1}{2}\left[\frac{y(x+h) - y(x)}{h} + \frac{y(x) - y(x-h)}{h}\right] \tag{1.35}$$

or

$$y'(x) \approx \frac{y(x+h) - y(x-h)}{2h}. \tag{1.36}$$

This formula, which is more accurate than either the forward or backward difference formulas, is the *central finite difference approximation* to the derivative.

Similar approximations can be derived for second derivatives. Using forward differences,

$$\begin{aligned} y''(x) &\approx \frac{y'(x+h) - y'(x)}{h} \\ &\approx \frac{y(x+2h) - y(x+h)}{h^2} - \frac{y(x+h) - y(x)}{h^2} \\ &= \frac{y(x+2h) - 2y(x+h) + y(x)}{h^2}. \end{aligned} \tag{1.37}$$

This formula, which involves three points forward of x, is the forward difference approximation to the second derivative. Similarly, using backward differences,

$$\begin{aligned} y''(x) &\approx \frac{y'(x) - y'(x-h)}{h} \\ &\approx \frac{y(x) - y(x-h)}{h^2} - \frac{y(x-h) - y(x-2h)}{h^2} \\ &= \frac{y(x) - 2y(x-h) + y(x-2h)}{h^2}. \end{aligned} \tag{1.38}$$

This formula, which involves three points backward of x, is the backward difference approximation to the second derivative. The central finite difference approximation to the second derivative uses instead the three points which bracket x:

$$y''(x) \approx \frac{y(x+h) - 2y(x) + y(x-h)}{h^2}. \tag{1.39}$$

This last result can also be obtained by using forward differences for the second derivative followed by backward differences for the first derivatives, or *vice versa*.

The central difference formula for second derivatives can alternatively be derived using Taylor series expansions:

$$y(x+h) = y(x) + hy'(x) + \frac{h^2}{2}y''(x) + \frac{h^3}{6}y'''(x) + O(h^4). \tag{1.40}$$

Similarly, by replacing h by $-h$,

$$y(x-h) = y(x) - hy'(x) + \frac{h^2}{2}y''(x) - \frac{h^3}{6}y'''(x) + O(h^4). \tag{1.41}$$

The addition of these two equations yields

$$y(x+h) + y(x-h) = 2y(x) + h^2y''(x) + O(h^4) \tag{1.42}$$

or

$$y''(x) = \frac{y(x+h) - 2y(x) + y(x-h)}{h^2} + O(h^2), \tag{1.43}$$

which, because of the error term, shows that the formula is second-order accurate.

1.6 Boundary Value Problems

The techniques for initial value problems (IVPs) are, in general, not directly applicable to boundary value problems (BVPs). Consider the BVP

$$\begin{cases} y''(x) = f(x, y, y'), & a \le x \le b \\ y(a) = \eta_1, \quad y(b) = \eta_2. \end{cases} \tag{1.44}$$

This equation could be nonlinear, depending on f. The methods used for IVPs started at one end ($x = a$) and computed the solution step by step for increasing x. For a BVP, not enough information is given at *either* endpoint to allow a step-by-step solution.

Consider first a special case of Eq. 1.44 for which the right-hand side depends only on x and y:

$$\begin{cases} y''(x) = f(x, y), & a \le x \le b \\ y(a) = \eta_1, \quad y(b) = \eta_2. \end{cases} \tag{1.45}$$

Subdivide the interval (a, b) into n equal subintervals:

$$h = \frac{b - a}{n}, \tag{1.46}$$

in which case

$$x_k = a + kh, \quad k = 0, 1, 2, \ldots, n. \tag{1.47}$$

Let y_k denote the numerical approximation to the exact solution at x_k. That is,

$$y_k \approx y(x_k). \tag{1.48}$$

Then, if we use a central difference approximation to the second derivative in Eq. 1.45, the ODE can be approximated by

$$\frac{y(x_{k-1}) - 2y(x_k) + y(x_{k+1})}{h^2} \approx f(x_k, y_k), \tag{1.49}$$

which suggests the difference equation

$$y_{k-1} - 2y_k + y_{k+1} = h^2 f(x_k, y_k), \quad k = 1, 2, 3, \ldots, n - 1. \tag{1.50}$$

Since this system of equations has $n - 1$ equations in $n + 1$ unknowns, the two boundary conditions are needed to obtain a nonsingular system:

$$y_0 = \eta_1, \quad y_n = \eta_2. \tag{1.51}$$

The resulting system is thus

$$\begin{array}{rcrcrcrcl} -2y_1 & + & y_2 & & & & & = & -\eta_1 + h^2 f(x_1, y_1) \\ y_1 & - & 2y_2 & + & y_3 & & & = & h^2 f(x_2, y_2) \\ & & y_2 & - & 2y_3 & + & y_4 & = & h^2 f(x_3, y_3) \\ & & & & y_3 & - & 2y_4 + y_5 & = & h^2 f(x_4, y_4) \\ & & & & & & \vdots & & \\ & & & & y_{n-2} & - & 2y_{n-1} & = & -\eta_2 + h^2 f(x_{n-1}, y_{n-1}), \end{array} \tag{1.52}$$

which is a tridiagonal system of $n - 1$ equations in $n - 1$ unknowns. This system is linear or nonlinear, depending on f.

1.6.1 Example

Consider

$$\begin{cases} y'' = -y(x), \quad 0 \le x \le \pi/2 \\ y(0) = 1, \quad y(\pi/2) = 0. \end{cases} \tag{1.53}$$

In Eq. 1.45, $f(x, y) = -y$, $\eta_1 = 1$, $\eta_2 = 0$. Thus, the right-hand side of the ith equation in Eq. 1.52 has $-h^2 y_i$, which can be moved to the left-hand side to yield the system

$$\begin{array}{rcrcrcrcl} -(2-h^2)y_1 & + & y_2 & & & & & = & -1 \\ y_1 & - & (2-h^2)y_2 & + & y_3 & & & = & 0 \\ & & y_2 & - & (2-h^2)y_3 & + & y_4 & = & 0 \\ & & & & & & \vdots & & \\ & & & & y_{n-2} & - & (2-h^2)y_{n-1} & = & 0. \end{array} \tag{1.54}$$

We first solve this tridiagonal system of simultaneous equations with $n = 8$ (i.e., $h = \pi/16$), and compare with the exact solution $y(x) = \cos x$:

k	x_k	y_k	Exact $y(x_k)$	Absolute Error	% Error
0	0	1	1	0	0
1	0.1963495	0.9812186	0.9807853	0.0004334	0.0441845
2	0.3926991	0.9246082	0.9238795	0.0007287	0.0788715
3	0.5890486	0.8323512	0.8314696	0.0008816	0.1060315
4	0.7853982	0.7080045	0.7071068	0.0008977	0.1269565
5	0.9817477	0.5563620	0.5555702	0.0007917	0.1425085
6	1.1780972	0.3832699	0.3826834	0.0005865	0.1532604
7	1.3744468	0.1954016	0.1950903	0.0003113	0.1595767
8	1.5707963	0	0	0	0

We then solve this system with $n = 40$ ($h = \pi/80$):

k	x_k	y_k	Exact $y(x_k)$	Absolute Error	% Error
0	0	1	1	0	0
5	0.1963495	0.9808025	0.9807853	0.0000172	0.0017571
10	0.3926991	0.9239085	0.9238795	0.0000290	0.0031363
15	0.5890486	0.8315047	0.8314696	0.0000351	0.0042161
20	0.7853982	0.7071425	0.7071068	0.0000357	0.0050479
25	0.9817477	0.5556017	0.5555702	0.0000315	0.0056660
30	1.1780972	0.3827068	0.3826834	0.0000233	0.0060933
35	1.3744468	0.1951027	0.1950903	0.0000124	0.0063443
40	1.5707963	0	0	0	0

Notice that a mesh refinement by a factor of 5 has reduced the error by a factor of about 25. This behavior is typical of a numerical method which is second-order accurate.

1.6.2 Solving Tridiagonal Systems

Tridiagonal systems are particularly easy (and fast) to solve using Gaussian elimination. It is convenient to solve such systems using the following notation:

$$\left.\begin{array}{rcrcrcrcl} d_1x_1 & + & u_1x_2 & & & & & = & b_1 \\ l_2x_1 & + & d_2x_2 & + & u_2x_3 & & & = & b_2 \\ & & l_3x_2 & + & d_3x_3 & + & u_3x_4 & = & b_3 \\ & & & & & & & \vdots & \\ & & & & l_nx_{n-1} & + & d_nx_n & = & b_n, \end{array}\right\} \tag{1.55}$$

where d_i, u_i, and l_i are, respectively, the diagonal, upper, and lower matrix entries in Row i. All coefficients can now be stored in three one-dimensional arrays, `D`$(\cdot)$, `U`$(\cdot)$, and `L`$(\cdot)$, instead of a full two-dimensional array `A(I,J)`. The solution algorithm (reduction to upper triangular form by Gaussian elimination followed by back-solving) can now be summarized as follows:

1. For $k = 1, 2, \ldots, n-1$: [k = pivot row]
   (a) $m = -l_{k+1}/d_k$ [m = multiplier needed to annihilate term below]
   (b) $d_{k+1} = d_{k+1} + mu_k$ [new diagonal entry in next row]
   (c) $b_{k+1} = b_{k+1} + mb_k$ [new rhs in next row]
2. $x_n = b_n/d_n$ [start of back-solve]
3. For $k = n-1, n-2, \ldots, 1$: [back-solve loop]
   (a) $x_k = (b_k - u_kx_{k+1})/d_k$

Tridiagonal systems arise in a variety of applications, including the Crank-Nicolson finite difference method for solving parabolic partial differential equations.

1.7 Shooting Methods

Shooting methods provide a way to convert a boundary value problem to a trial-and-error initial value problem. It is useful to have additional ways to solve BVPs, particularly if the equations are nonlinear.

Consider the following two-point BVP:

$$\begin{cases} y'' = f(x, y, y'), & a \le x \le b \\ y(a) = A \\ y(b) = B. \end{cases} \tag{1.56}$$

To solve this problem using the shooting method, we compute solutions of the IVP

$$\begin{cases} y'' = f(x, y, y'), & x \ge a \\ y(a) = A \\ y'(a) = M \end{cases} \tag{1.57}$$

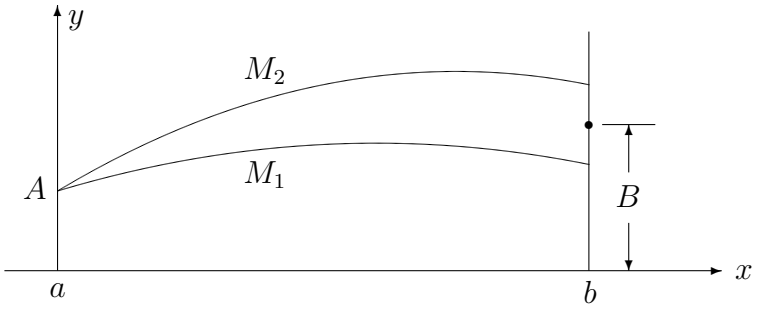


Figure 4: The Shooting Method.

for various values of M (the slope at the left end of the domain) until two solutions, one with $y(b) < B$ and the other with $y(b) > B$, have been found (Fig. 4). The initial slope M can then be interpolated until a solution is found (i.e., $y(b) = B$).

2 Partial Differential Equations

A *partial differential equation* (PDE) is an equation that involves an unknown function (the dependent variable) and some of its partial derivatives with respect to two or more independent variables. An nth-order equation has the highest order derivative of order n.

2.1 Classical Equations of Mathematical Physics

1. Laplace's equation (the potential equation)

$$\nabla^2 \phi = 0 \tag{2.1}$$

In Cartesian coordinates, the vector operator *del* is defined as

$$\nabla = \mathbf{e}_x \frac{\partial}{\partial x} + \mathbf{e}_y \frac{\partial}{\partial y} + \mathbf{e}_z \frac{\partial}{\partial z}. \tag{2.2}$$

∇^2 is referred to as the Laplacian operator and given by

$$\nabla^2 = \nabla \cdot \nabla = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}. \tag{2.3}$$

Thus, Laplace's equation in Cartesian coordinates is

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} + \frac{\partial^2 \phi}{\partial z^2} = 0. \tag{2.4}$$

In cylindrical coordinates, the Laplacian is

$$\nabla^2 \phi = \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial \phi}{\partial r} \right) + \frac{1}{r^2} \frac{\partial^2 \phi}{\partial \theta^2} + \frac{\partial^2 \phi}{\partial z^2}. \tag{2.5}$$

Laplace's equation arises in incompressible fluid flow (in which case ϕ is the velocity potential), gravitational potential problems, electrostatics, magnetostatics, steady-state

heat conduction with no sources (in which case ϕ is the temperature), and torsion of bars in elasticity (in which case $\phi(x, y)$ is the warping function). Functions which satisfy Laplace's equation are referred to as *harmonic* functions.

2. Poisson's equation

$$\nabla^2\phi + g = 0 \tag{2.6}$$

This equation arises in steady-state heat conduction with distributed sources (ϕ = temperature) and torsion of bars in elasticity (in which case $\phi(x, y)$ is the stress function).

3. wave equation

$$\nabla^2\phi = \frac{1}{c^2}\ddot{\phi} \tag{2.7}$$

In this equation, dots denote time derivatives, e.g.,

$$\ddot{\phi} = \frac{\partial^2\phi}{\partial t^2}, \tag{2.8}$$

and c is the speed of propagation. The wave equation arises in several physical situations:

(a) *transverse vibrations of a string*
For this one-dimensional problem, $\phi = \phi(x, t)$ is the transverse displacement of the string, and

$$c = \sqrt{\frac{T}{\rho A}}, \tag{2.9}$$

where T is the string tension, ρ is the density of the string material, and A is the cross-sectional area of the string. The denominator ρA is mass per unit length.

(b) *longitudinal vibrations of a bar*
For this one-dimensional problem, $\phi = \phi(x, t)$ represents the longitudinal displacement, and

$$c = \sqrt{\frac{E}{\rho}}, \tag{2.10}$$

where E and ρ are, respectively, the modulus of elasticity and density of the bar material.

(c) *transverse vibrations of a membrane*
For this two-dimensional problem, $\phi = \phi(x, y, t)$ is the transverse displacement of the membrane (e.g., drum head), and

$$c = \sqrt{\frac{T}{m}}, \tag{2.11}$$

where T is the tension per unit length, m is the mass per unit area (i.e., $m = \rho t$), and t is the membrane thickness.

(d) *acoustics*
For this three-dimensional problem, $\phi = \phi(x, y, z, t)$ is the fluid pressure or velocity potential, and c is the speed of sound, where

$$c = \sqrt{\frac{B}{\rho}}, \tag{2.12}$$

where $B = \rho c^2$ is the fluid bulk modulus, and ρ is the density.

4. Helmholtz equation (reduced wave equation)

$$\nabla^2 \phi + k^2 \phi = 0 \tag{2.13}$$

The Helmholtz equation is the time-harmonic form of the wave equation, in which interest is restricted to functions which vary sinusoidally in time. To obtain the Helmholtz equation, we substitute

$$\phi(x, y, z, t) = \phi_0(x, y, z) \cos \omega t \tag{2.14}$$

into the wave equation, Eq. 2.7, to obtain

$$\nabla^2 \phi_0 \cos \omega t = -\frac{\omega^2}{c^2} \phi_0 \cos \omega t. \tag{2.15}$$

If we define the wave number $k = \omega/c$, this equation becomes

$$\nabla^2 \phi_0 + k^2 \phi_0 = 0. \tag{2.16}$$

With the understanding that the unknown depends only on the spacial variables, the subscript is unnecessary, and we obtain the Helmholtz equation, Eq. 2.13. This equation arises in steady-state (time-harmonic) situations involving the wave equation, e.g., steady-state acoustics.

5. heat equation

$$\nabla \cdot (k \nabla \phi) + Q = \rho c \dot{\phi} \tag{2.17}$$

In this equation, ϕ represents the temperature T, k is the thermal conductivity, Q is the internal heat generation per unit volume per unit time, ρ is the material density, and c is the material specific heat (the heat required per unit mass to raise the temperature by one degree). The thermal conductivity k is defined by Fourier's law of heat conduction:

$$\hat{q}_x = -kA \frac{dT}{dx}, \tag{2.18}$$

where $\hat{q}_x$ is the rate of heat conduction (energy per unit time) with typical units J/s or BTU/hr, and A is the area through which the heat flows. Alternatively, Fourier's law is written

$$q_x = -k \frac{dT}{dx}, \tag{2.19}$$

where q_x is energy per unit time per unit area (with typical units J/(s·m^2). There are several special cases of the heat equation of interest:

(a) homogeneous material (k = constant):

$$k\nabla^2\phi + Q = \rho c\dot{\phi} \tag{2.20}$$

(b) homogeneous material, steady-state (time-independent):

$$\nabla^2\phi = -\frac{Q}{k} \quad \text{(Poisson's equation)} \tag{2.21}$$

(c) homogeneous material, steady-state, no sources ($Q = 0$):

$$\nabla^2\phi = 0 \quad \text{(Laplace's equation)} \tag{2.22}$$

2.2 Classification of Partial Differential Equations

Of the classical PDEs summarized in the preceding section, some involve time, and some don't, so presumably their solutions would exhibit fundamental differences. Of those that involve time (wave and heat equations), the order of the time derivative is different, so the fundamental character of their solutions may also differ. Both these speculations turn out to be true.

Consider the general, second-order, linear partial differential equation in two variables

$$Au_{xx} + Bu_{xy} + Cu_{yy} + Du_x + Eu_y + Fu = G, \tag{2.23}$$

where the coefficients are functions of the independent variables x and y (i.e., $A = A(x, y)$, $B = B(x, y)$, etc.), and we have used subscripts to denote partial derivatives, e.g.,

$$u_{xx} = \frac{\partial^2 u}{\partial x^2}. \tag{2.24}$$

The quantity $B^2 - 4AC$ is referred to as the *discriminant* of the equation. The behavior of the solution of Eq. 2.23 depends on the sign of the discriminant according to the following table:

$B^2 - 4AC$	Equation Type	Typical Physics Described
< 0	Elliptic	Steady-state phenomena
$= 0$	Parabolic	Heat flow and diffusion processes
> 0	Hyperbolic	Vibrating systems and wave motion

The names elliptic, parabolic, and hyperbolic arise from the analogy with the conic sections in analytic geometry.

Given these definitions, we can classify the common equations of mathematical physics already encountered as follows:

Name	Eq. Number	Eq. in Two Variables	A, B, C	Type
Laplace	Eq. 2.1	$u_{xx} + u_{yy} = 0$	$A = C = 1, B = 0$	Elliptic
Poisson	Eq. 2.6	$u_{xx} + u_{yy} = -g$	$A = C = 1, B = 0$	Elliptic
Wave	Eq. 2.7	$u_{xx} - u_{yy}/c^2 = 0$	$A = 1, C = -1/c^2, B = 0$	Hyperbolic
Helmholtz	Eq. 2.13	$u_{xx} + u_{yy} + k^2 u = 0$	$A = C = 1, B = 0$	Elliptic
Heat	Eq. 2.17	$ku_{xx} - \rho c u_y = -Q$	$A = k, B = C = 0$	Parabolic

In the wave and heat equations in the above table, y represents the time variable. The behavior of the solutions of equations of different types differs. Elliptic equations characterize static (time-independent) situations, and the other two types of equations characterize time-dependent situations.

2.3 Transformation to Nondimensional Form

It is often convenient, when solving equations, to transform the equation to a nondimensional form. Consider, for example, the one-dimensional wave equation

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{c^2}\frac{\partial^2 u}{\partial t^2}, \tag{2.25}$$

where u is displacement (of dimension length), t is time, and c is the speed of propagation (of dimension length/time). Let L represent some characteristic length associated with the problem. We define the nondimensional variables

$$\bar{x} = \frac{x}{L}, \quad \bar{u} = \frac{u}{L}, \quad \bar{t} = \frac{ct}{L}, \tag{2.26}$$

in which case the derivatives in Eq. 2.25 become

$$\frac{\partial u}{\partial x} = \frac{\partial (L\bar{u})}{\partial (L\bar{x})} = \frac{\partial \bar{u}}{\partial \bar{x}}, \tag{2.27}$$

$$\frac{\partial^2 u}{\partial x^2} = \frac{\partial}{\partial x}\left(\frac{\partial u}{\partial x}\right) = \frac{\partial}{\partial x}\left(\frac{\partial \bar{u}}{\partial \bar{x}}\right) = \frac{\partial}{\partial \bar{x}}\left(\frac{\partial \bar{u}}{\partial \bar{x}}\right)\frac{d\bar{x}}{dx} = \frac{1}{L}\frac{\partial^2 \bar{u}}{\partial \bar{x}^2}, \tag{2.28}$$

$$\frac{\partial u}{\partial t} = \frac{\partial (L\bar{u})}{\partial (L\bar{t}/c)} = c\frac{\partial \bar{u}}{\partial \bar{t}}, \tag{2.29}$$

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial}{\partial t}\left(\frac{\partial u}{\partial t}\right) = \frac{\partial}{\partial t}\left(c\frac{\partial \bar{u}}{\partial \bar{t}}\right) = c\frac{\partial}{\partial \bar{t}}\left(\frac{\partial \bar{u}}{\partial \bar{t}}\right)\frac{d\bar{t}}{dt} = c\frac{\partial^2 \bar{u}}{\partial \bar{t}^2}\frac{c}{L} = \frac{c^2}{L}\frac{\partial^2 \bar{u}}{\partial \bar{t}^2}. \tag{2.30}$$

Thus, from Eq. 2.25,

$$\frac{1}{L}\frac{\partial^2 \bar{u}}{\partial \bar{x}^2} = \frac{1}{c^2}\cdot\frac{c^2}{L}\frac{\partial^2 \bar{u}}{\partial \bar{t}^2} \tag{2.31}$$

or

$$\frac{\partial^2 \bar{u}}{\partial \bar{x}^2} = \frac{\partial^2 \bar{u}}{\partial \bar{t}^2}. \tag{2.32}$$

This is the nondimensional wave equation.

This last equation can also be obtained more easily by direct substitution of Eq. 2.26 into Eq. 2.25 and factoring out the constants:

$$\frac{\partial^2 (L\bar{u})}{\partial (L\bar{x})^2} = \frac{1}{c^2}\frac{\partial^2 (L\bar{u})}{\partial (L\bar{t}/c)^2} \tag{2.33}$$

or

$$\frac{L}{L^2}\frac{\partial^2 \bar{u}}{\partial \bar{x}^2} = \frac{1}{c^2}\frac{c^2 L}{L^2}\frac{\partial^2 \bar{u}}{\partial \bar{t}^2}. \tag{2.34}$$

3 Finite Difference Solution of Partial Differential Equations

3.1 Parabolic Equations

Consider the boundary-initial value problem (BIVP)

$$\begin{cases} u_{xx} = \dfrac{1}{c}\, u_t, \quad u = u(x,t), \quad 0 < x < 1, \quad t > 0 \\ u(0,t) = u(1,t) = 0 \quad \text{(boundary conditions)} \\ u(x,0) = f(x) \quad \text{(initial condition)}, \end{cases} \tag{3.1}$$

where c is a constant. This problem represents transient heat conduction in a rod with the ends held at zero temperature and an initial temperature profile $f(x)$.

To solve this problem numerically, we discretize x and t such that

$$\begin{aligned} x_i &= ih, \quad i = 0, 1, 2, \ldots \\ t_j &= jk, \quad j = 0, 1, 2, \ldots . \end{aligned} \tag{3.2}$$

3.1.1 Explicit Finite Difference Method

Let $u_{i,j}$ be the numerical approximation to $u(x_i, t_j)$. We approximate u_t with the forward finite difference

$$u_t \approx \frac{u_{i,j+1} - u_{i,j}}{k} \tag{3.3}$$

and u_{xx} with the central finite difference

$$u_{xx} \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2}. \tag{3.4}$$

The finite difference approximation to the PDE is then

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} = \frac{u_{i,j+1} - u_{i,j}}{ck}. \tag{3.5}$$

Define the parameter r as

$$r = \frac{ck}{h^2} = \frac{c\,\Delta t}{(\Delta x)^2}, \tag{3.6}$$

in which case Eq. 3.5 becomes

$$u_{i,j+1} = ru_{i-1,j} + (1 - 2r)u_{i,j} + ru_{i+1,j}. \tag{3.7}$$

The domain of the problem and the mesh are illustrated in Fig. 5. Eq. 3.7 is a recursive relationship giving u in a given row (time) in terms of three consecutive values of u in the row below (one time step earlier). This equation is referred to as an *explicit* formula since one unknown value can be found directly in terms of several other known values. The recursive relationship can also be sketched with the stencil shown in Fig. 6. For example, for $r = 1/10$, we have the stencil shown in Fig. 7. That is, for $r = 1/10$, the solution (temperature) at

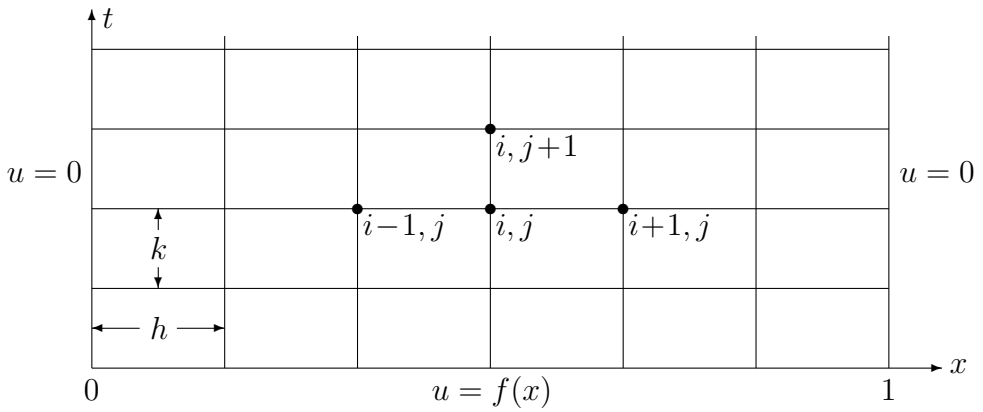


Figure 5: Mesh for 1-D Heat Equation.

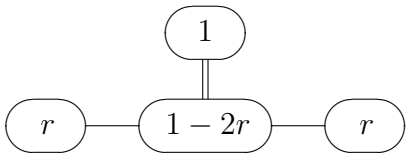


Figure 6: Heat Equation Stencil for Explicit Finite Difference Algorithm.

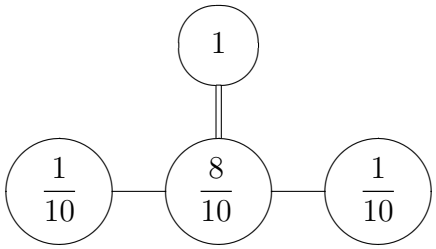


Figure 7: Heat Equation Stencil for $r = 1/10$.

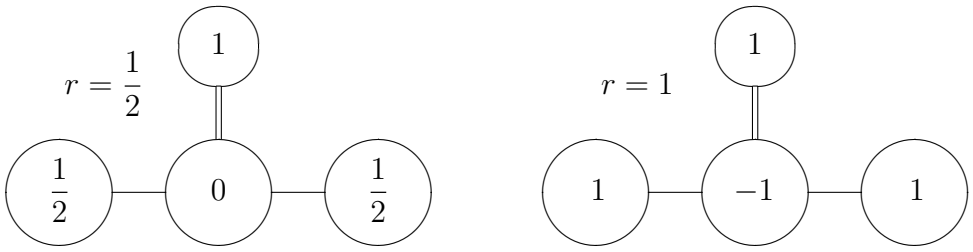


Figure 8: Heat Equation Stencils for $r = 1/2$ and $r = 1$.

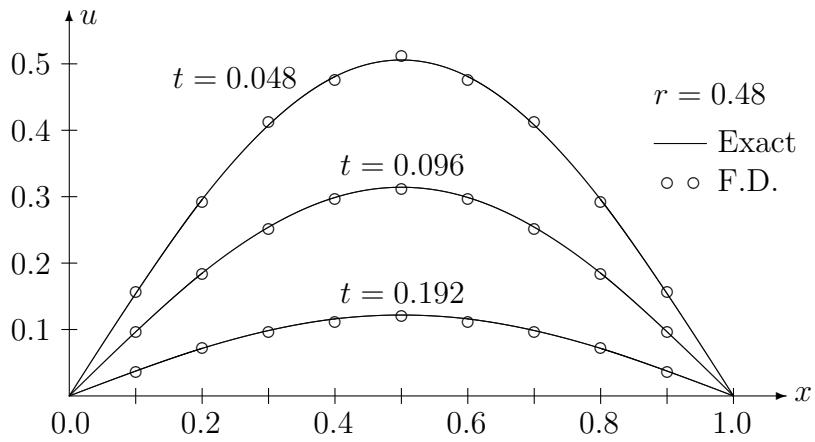


Figure 9: Explicit Finite Difference Solution With $r = 0.48$.

the new point depends on the three points at the previous time step with a 1-8-1 weighting.

Notice that, if $r = 1/2$, the solution at the new point is *independent* of the closest point, as illustrated in Fig. 8. For $r > 1/2$ (e.g., $r = 1$), the new point depends *negatively* on the closest point (Fig. 8), which is counter-intuitive. It can be shown that, for a stable solution, $0 < r \leq 1/2$. An unstable solution is one for which small errors grow rather than decay as the solution evolves.

The instability which occurs for $r > 1/2$ can be illustrated with the following example. Consider the boundary-initial value problem (in nondimensional form)

$$\begin{aligned} &u_{xx} = u_t, \quad 0 < x < 1, \quad u = u(x,t) \\ &u(0,t) = u(1,t) = 0 \\ &u(x,0) = f(x) = \begin{cases} 2x, & 0 \leq x \leq 1/2 \\ 2(1-x), & 1/2 \leq x \leq 1. \end{cases} \end{aligned} \tag{3.8}$$

The physical problem is to compute the temperature history $u(x,t)$ for a bar with a prescribed initial temperature distribution $f(x)$, no internal heat sources, and zero temperature prescribed at both ends. We solve this problem using the explicit finite difference algorithm with $h = \Delta x = 0.1$ and $k = \Delta t = rh^2 = r(\Delta x)^2$ for two different values of r: $r = 0.48$ and $r = 0.52$. The two numerical solutions (Figs. 9 and 10) are compared with the analytic solution

$$u(x,t) = \sum_{n=1}^{\infty} \frac{8}{(n\pi)^2} \sin\frac{n\pi}{2} (\sin n\pi x)\, e^{-(n\pi)^2 t}, \tag{3.9}$$

which can be obtained by the technique of separation of variables. The instability for $r > 1/2$ can be clearly seen in Fig. 10. Thus, a disadvantage of this explicit method is that a small time step Δt must be used to maintain stability. This disadvantage will be removed with the Crank-Nicolson algorithm.

3.1.2 Crank-Nicolson Implicit Method

The Crank-Nicolson method is a stable algorithm which allows a larger time step than could be used in the explicit method. In fact, Crank-Nicolson's stability does not depend on the parameter r.

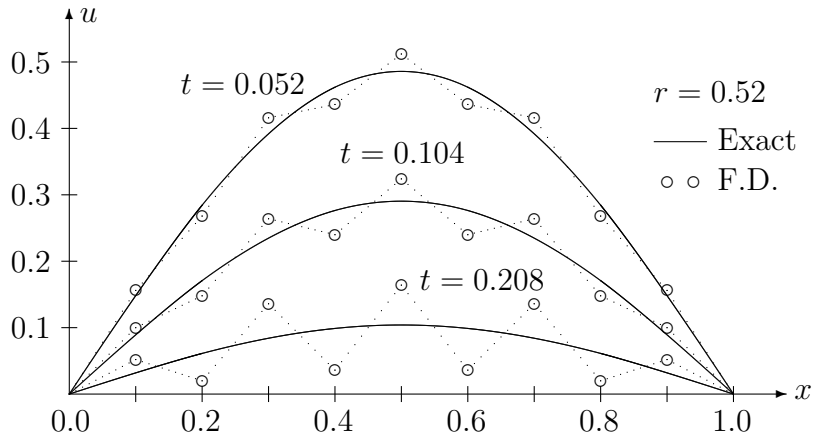


Figure 10: Explicit Finite Difference Solution With $r = 0.52$.

The basis for the Crank-Nicolson algorithm is writing the finite difference equation at a mid-level in time: $i, j + \frac{1}{2}$. The finite difference x derivative at $j + \frac{1}{2}$ is computed as the average of the two central difference time derivatives at j and $j+1$. Consider again the PDE of Eq. 3.1:

$$\begin{cases} u_{xx} = \dfrac{1}{c}\, u_t, \quad u = u(x,t) \\ u(0,t) = u(1,t) = 0 \quad \text{(boundary conditions)} \\ u(x,0) = f(x) \quad \text{(initial condition)}. \end{cases} \tag{3.10}$$

The PDE is approximated numerically by

$$\frac{1}{2}\left[\frac{u_{i+1,j+1} - 2u_{i,j+1} + u_{i-1,j+1}}{h^2} + \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2}\right] = \frac{u_{i,j+1} - u_{i,j}}{ck}, \tag{3.11}$$

where the right-hand side is a central difference approximation to the time derivative at the middle point $j + \frac{1}{2}$. We again define the parameter r as

$$r = \frac{ck}{h^2} = \frac{c\,\Delta t}{(\Delta x)^2}, \tag{3.12}$$

and rearrange Eq. 3.11 with all $j + 1$ terms on the left-hand side:

$$-ru_{i-1,j+1} + 2(1+r)u_{i,j+1} - ru_{i+1,j+1} = ru_{i-1,j} + 2(1-r)u_{i,j} + ru_{i+1,j}. \tag{3.13}$$

This formula is called the *Crank-Nicolson* algorithm.

Fig. 11 shows the points involved in the Crank-Nicolson scheme. If we start at the bottom row ($j = 0$) and move up, the right-hand side values of Eq. 3.13 are known, and the left-hand side values of that equation are unknown. To get the process started, let $j = 0$, and write the C-N equation for each $i = 1, 2, \ldots, N$ to obtain N simultaneous equations in N unknowns, where N is the number of *interior* mesh points on the row. (The boundary points, with known values, are excluded.) This system of equations is a tridiagonal system, since each equation has three consecutive nonzeros centered around the diagonal. To advance in time,

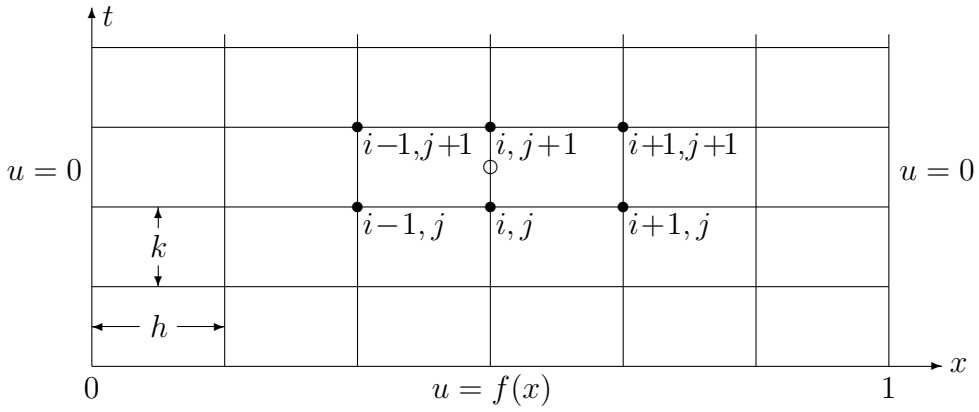


Figure 11: Mesh for Crank-Nicolson.

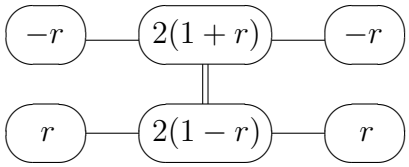


Figure 12: Stencil for Crank-Nicolson Algorithm.

we then increment j to $j = 1$, and solve a new system of equations. An approach which requires the solution of simultaneous equations is called an *implicit* algorithm. A sketch of the C-N stencil is shown in Fig. 12.

Note that the coefficient matrix of the C-N system of equations does not change from step to step. Thus, one could compute and save the LU factors of the coefficient matrix, and merely do the forward-backward substitution (FBS) at each new time step, thus speeding up the calculation. This speedup would be particularly significant in higher dimensional problems, where the coefficient matrix is no longer tridiagonal.

It can be shown that the C-N algorithm is stable for any r, although better accuracy results from a smaller r. A smaller r corresponds to a smaller time step size (for a fixed spacial mesh). C-N also gives better accuracy than the explicit approach for the same r.

3.1.3 Derivative Boundary Conditions

Consider the boundary-initial value problem

$$\begin{cases} u_{xx} = \dfrac{1}{c}\, u_t, \quad u = u(x,t) \\ u(0,t) = 0, \quad u_x(1,t) = g(t) \quad \text{(boundary conditions)} \\ u(x,0) = f(x) \quad \text{(initial condition)}. \end{cases} \tag{3.14}$$

The only difference between this problem and the one considered earlier in Eq. 3.1 is the right-hand side boundary condition, which now involves a derivative (a *Neumann* boundary condition).

Assume a mesh labeling as shown in Fig. 13. We introduce extra "phantom" points to the right of the boundary (outside the domain). Consider boundary Point 25, for example,

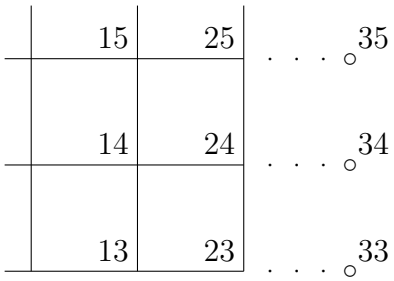


Figure 13: Treatment of Derivative Boundary Conditions.

and assume we use an explicit finite difference algorithm. Since u_{25} is not known, we must write the finite difference equation for u_{25}:

$$u_{25} = ru_{14} + (1-2r)u_{24} + ru_{34}. \tag{3.15}$$

On the other hand, a central finite difference approximation to the x derivative at Point 24 is

$$\frac{u_{34} - u_{14}}{2h} = g_{24}. \tag{3.16}$$

The phantom variable u_{34} can then be eliminated from the last two equations to yield a new equation for the boundary point u_{25}:

$$u_{25} = 2ru_{14} + (1-2r)u_{24} + 2rhg_{24}. \tag{3.17}$$

3.2 Hyperbolic Equations

3.2.1 The d'Alembert Solution of the Wave Equation

Before addressing the finite difference solution of hyperbolic equations, we review some background material on such equations.

The time-dependent transverse response of an infinitely long string satisfies the one-dimensional wave equation with nonzero initial displacement and velocity specified:

$$\begin{cases} \dfrac{\partial^2 u}{\partial x^2} = \dfrac{1}{c^2}\dfrac{\partial^2 u}{\partial t^2}, & -\infty < x < \infty, \quad t > 0, \\ u(x,0) = f(x), \quad \dfrac{\partial u(x,0)}{\partial t} = g(x) \\ \lim\limits_{x \to \pm\infty} u(x,t) = 0, \end{cases} \tag{3.18}$$

where x is distance along the string, t is time, $u(x,t)$ is the transverse displacement, $f(x)$ is the initial displacement, $g(x)$ is the initial velocity, and the constant c is given by

$$c = \sqrt{\frac{T}{\rho A}}, \tag{3.19}$$

where T is the tension (force) in the string, ρ is the density of the string material, and A is the cross-sectional area of the string. Note that c has the dimension of velocity. This

equation assumes that all motion is vertical and that the displacement u and its slope $\partial u/\partial x$ are both small.

It can be shown by direct substitution into Eq. 3.18 that the solution of this system is

$$u(x,t) = \frac{1}{2}[f(x-ct) + f(x+ct)] + \frac{1}{2c}\int_{x-ct}^{x+ct} g(\tau)\,d\tau. \tag{3.20}$$

The differentiation of the integral in Eq. 3.20 is effected with the aid of Leibnitz's rule:

$$\frac{d}{dx}\int_{A(x)}^{B(x)} h(x,t)\,dt = \int_A^B \frac{\partial h(x,t)}{\partial x}\,dt + h(x,B)\frac{dB}{dx} - h(x,A)\frac{dA}{dx}. \tag{3.21}$$

Eq. 3.20 is known as the d'Alembert solution of the one-dimensional wave equation.

For the special case $g(x) = 0$ (zero initial velocity), the d'Alembert solution simplifies to

$$u(x,t) = \frac{1}{2}[f(x-ct) + f(x+ct)], \tag{3.22}$$

which may be interpreted as two waves, each equal to $f(x)/2$, which travel at speed c to the right and left, respectively. For example, the argument $x - ct$ remains constant if, as t increases, x also increases at speed c. Thus, the wave $f(x-ct)$ moves to the right (increasing x) with speed c without change of shape. Similarly, the wave $f(x+ct)$ moves to the left (decreasing x) with speed c without change of shape. The two waves [each equal to half the initial shape $f(x)$] travel in opposite directions from each other at speed c. If $f(x)$ is nonzero only for a small domain, then, after both waves have passed the region of initial disturbance, the string returns to its rest position.

For example, let $f(x)$, the initial displacement, be given by

$$f(x) = \begin{cases} -|x| + b, & |x| \le b, \\ 0, & |x| \ge b, \end{cases} \tag{3.23}$$

which is a triangular pulse of width $2b$ and height b (Fig. 14). For $t > 0$, half this pulse travels in opposite directions from the origin. For $t > b/c$, where c is the wave speed, the two half-pulses have completely separated, and the neighborhood of the origin has returned to rest.

For the special case $f(x) = 0$ (zero initial displacement), the d'Alembert solution simplifies to

$$u(x,t) = \frac{1}{2c}\int_{x-ct}^{x+ct} g(\tau)\,d\tau = \frac{1}{2}\left[G(x+ct) - G(x-ct)\right], \tag{3.24}$$

where

$$G(x) = \frac{1}{c}\int_{-\infty}^{x} g(\tau)\,d\tau. \tag{3.25}$$

Thus, similar to the initial displacement special case, this solution, Eq. 3.24, may be interpreted as the combination (difference, in this case) of two identical functions $G(x)/2$, one moving left and one moving right, each with speed c.

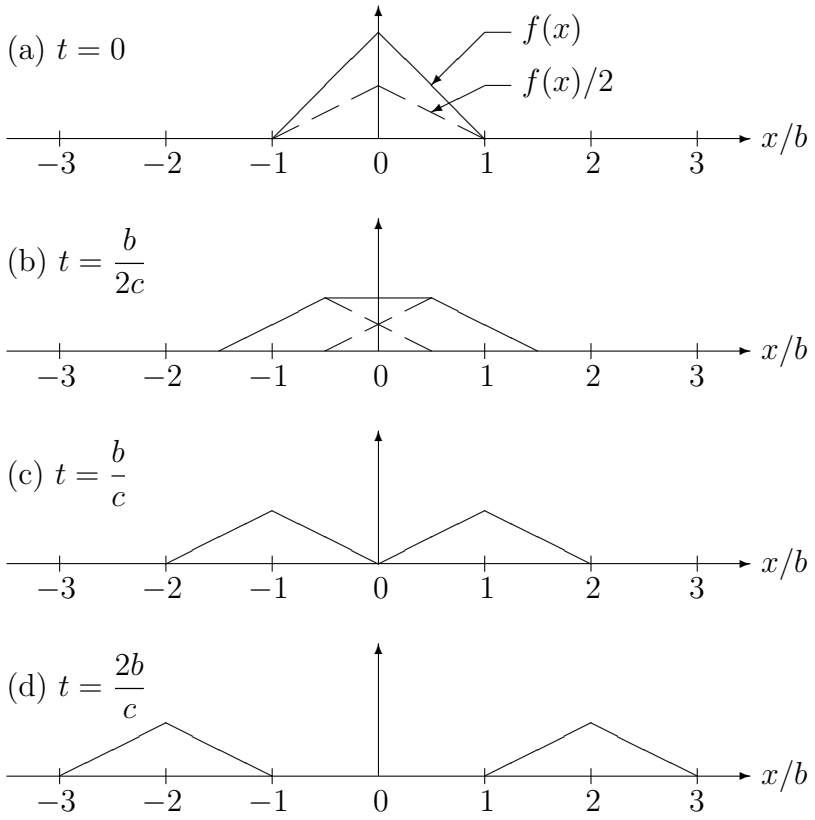


Figure 14: Propagation of Initial Displacement.

For example, let the initial velocity $g(x)$ be given by

$$g(x) = \begin{cases} Mc, & |x| < b, \\ 0, & |x| > b, \end{cases} \tag{3.26}$$

which is a rectangular pulse of width $2b$ and height Mc, where the constant M is the dimensionless Mach number, and c is the wave speed (Fig. 15). The travelling wave $G(x)$ is given by Eq. 3.25 as

$$G(x) = \begin{cases} 0, & x \le -b, \\ M(x+b), & -b \le x \le b, \\ 2Mb, & x \ge b. \end{cases} \tag{3.27}$$

That is, half this wave travels in opposite directions at speed c. Even though $g(x)$ is nonzero only near the origin, the travelling wave $G(x)$ is constant and nonzero for $x > b$. Thus, as time advances, the center section of the string reaches a state of rest, but not in its original position (Fig. 16).

From the preceding discussion, it is clear that disturbances travel with speed c. For an observer at some fixed location, initial displacements occurring elsewhere pass by after a finite time has elapsed, and then the string returns to rest in its original position. Nonzero initial velocity disturbances also travel at speed c, but, once having reached some location, will continue to influence the solution from then on.

Thus, the *domain of influence* of the data at $x = x_0$, say, on the solution consists of all points closer than ct (in either direction) to x_0, the location of the disturbance (Fig. 17).

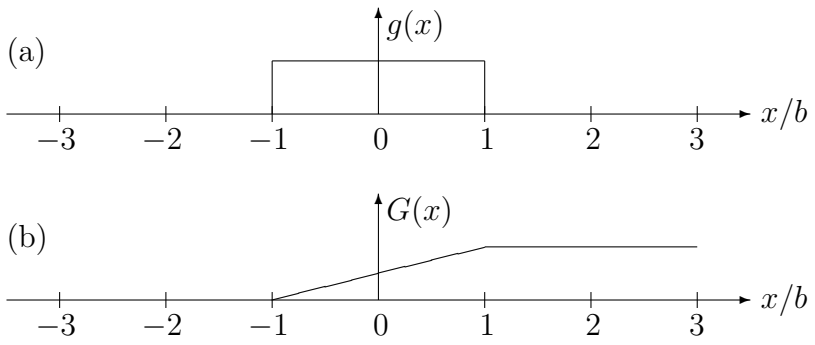


Figure 15: Initial Velocity Function.

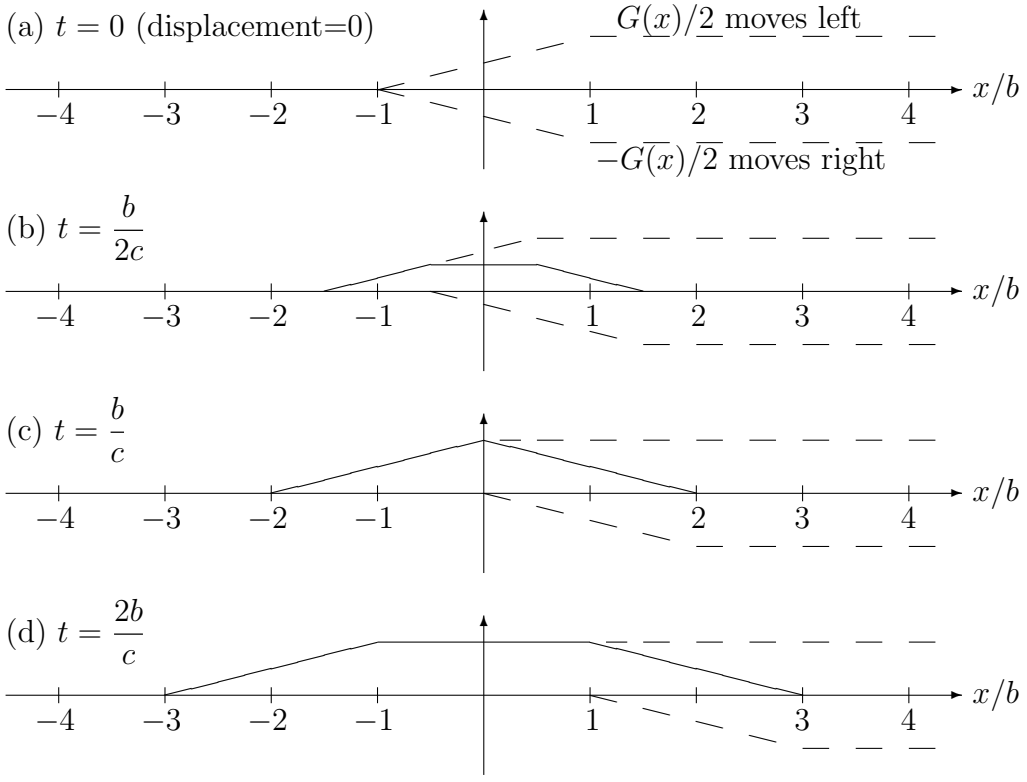
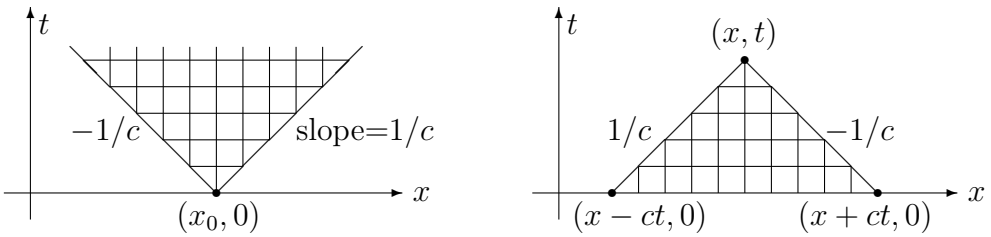


Figure 16: Propagation of Initial Velocity.



(a) Domain of Influence of Data on Solution.

(b) Domain of Dependence of Solution on Data.

Figure 17: Domains of Influence and Dependence.

Conversely, the *domain of dependence* of the solution on the initial data consists of all points within a distance ct of the solution point. That is, the solution at (x, t) depends on the initial data for all locations in the range $(x - ct, x + ct)$, which are the limits of integration in the d'Alembert solution, Eq. 3.20.

3.2.2 Finite Differences

From the preceding discussion of the d'Alembert solution, we see that hyperbolic equations involve wave motion. If the initial data are discontinuous (as, for example, in shocks), the most accurate and the most convenient approach for solving the equations is probably the method of characteristics. On the other hand, problems without discontinuities can probably be solved most conveniently using finite difference and finite element techniques. Here we consider finite differences.

Consider the boundary-initial value problem (BIVP)

$$\begin{cases} u_{xx} = \dfrac{1}{c^2}\, u_{tt}, \quad u = u(x,t), \quad 0 < x < a, \quad t > 0 \\ u(0,t) = u(a,t) = 0 \quad \text{(boundary conditions)} \\ u(x,0) = f(x), \quad u_t(x,0) = g(x) \quad \text{(initial conditions).} \end{cases} \tag{3.28}$$

This problem represents the transient (time-dependent) vibrations of a string fixed at the two ends with both initial displacement $f(x)$ and initial velocity $g(x)$ specified.

A central finite difference approximation to the PDE, Eq. 3.28, yields

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} = \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{c^2 k^2}. \tag{3.29}$$

We define the parameter

$$r = \frac{ck}{h} = \frac{c\,\Delta t}{\Delta x}, \tag{3.30}$$

and solve for $u_{i,j+1}$:

$$u_{i,j+1} = r^2 u_{i-1,j} + 2(1 - r^2)u_{i,j} + r^2 u_{i+1,j} - u_{i,j-1}. \tag{3.31}$$

Fig. 18 shows the mesh points involved in this recursive scheme. If we know the solution for

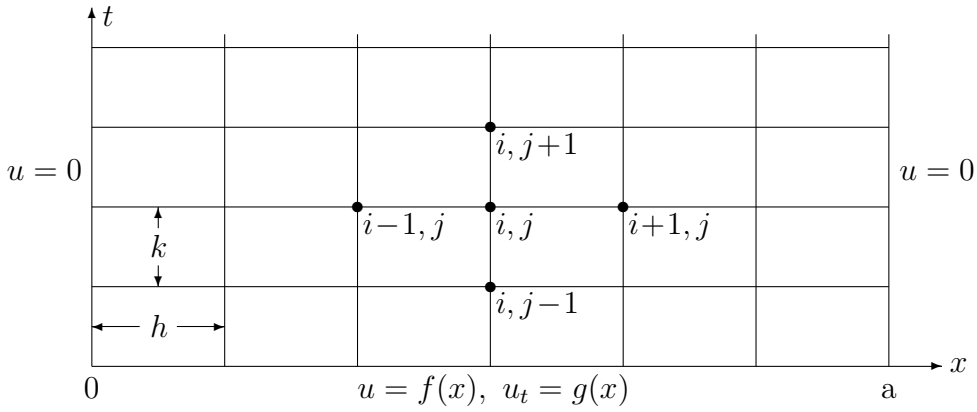


Figure 18: Mesh for Explicit Solution of Wave Equation.

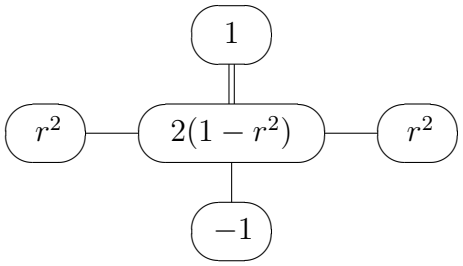


Figure 19: Stencil for Explicit Solution of Wave Equation.

all time values up to the jth time step, we can compute the solution $u_{i,j+1}$ at Step $j+1$ in terms of known quantities. Thus, this algorithm is an explicit algorithm. The corresponding stencil is shown in Fig. 19.

It can be shown that this finite difference algorithm is stable if $r \leq 1$ and unstable if $r > 1$. This stability condition is know as the Courant, Friedrichs, and Lewy (CFL) condition or simply the Courant condition. It can be further shown that a theoretically correct solution is obtained when $r = 1$, and that the accuracy of the solution decreases as the value of r decreases farther and farther below the value of 1. Thus, the time step size Δt should be chosen, if possible, so that $r = 1$. If that Δt is inconvenient for a particular calculation, Δt should be selected as large as possible without exceeding the stability limit of $r = 1$.

An intuitive rationale behind the stability requirement $r \leq 1$ can also be made using Fig. 20. If $r > 1$, the numerical domain of dependence (NDD) would be smaller than the actual domain of dependence (ADD) for the PDE, since NDD spreads by one mesh point at each level for earlier time. However, if NDD < ADD, the numerical solution would be independent of data outside NDD but inside ADD. That is, the numerical solution would ignore necessary information. Thus, to insure a stable solution, $r \leq 1$.

3.2.3 Starting Procedure for Explicit Algorithm

We note that the explicit finite difference scheme just described for the wave equation requires the numerical solution at two consecutive time steps to step forward to the next time. Thus, at $t = 0$, a special procedure is needed to advance from $j = 0$ to $j = 1$.

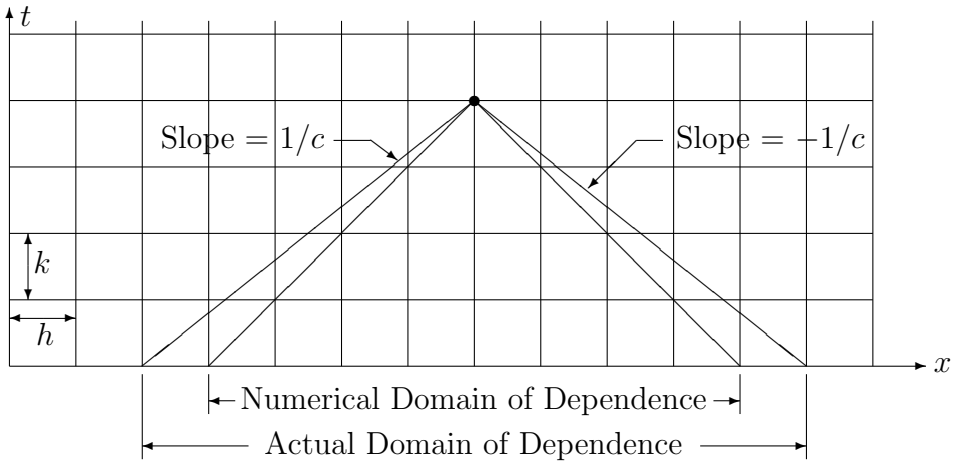


Figure 20: Domains of Dependence for $r > 1$.

The explicit finite difference algorithm is given in Eq. 3.31:

$$u_{i,j+1} = r^2 u_{i-1,j} + 2(1 - r^2)u_{i,j} + r^2 u_{i+1,j} - u_{i,j-1}. \tag{3.32}$$

To compute the solution at the end of the first time step, let $j = 0$:

$$u_{i,1} = r^2 u_{i-1,0} + 2(1 - r^2)u_{i,0} + r^2 u_{i+1,0} - u_{i,-1}, \tag{3.33}$$

where the right-hand side is known (from the initial condition) except for $u_{i,-1}$. However, we can write a central difference approximation to the first time derivative at $t = 0$:

$$\frac{u_{i,1} - u_{i,-1}}{2k} = g_i \tag{3.34}$$

or

$$u_{i,-1} = u_{i,1} - 2kg_i, \tag{3.35}$$

where g_i is the initial velocity $g(x)$ evaluated at the ith point, i.e., $g_i = g(x_i)$. If we substitute this last result into Eq. 3.33 (to eliminate $u_{i,-1}$), we obtain

$$u_{i,1} = r^2 u_{i-1,0} + 2(1 - r^2)u_{i,0} + r^2 u_{i+1,0} - u_{i,1} + 2kg_i \tag{3.36}$$

or

$$u_{i,1} = \frac{1}{2}r^2 u_{i-1,0} + (1 - r^2)u_{i,0} + \frac{1}{2}r^2 u_{i+1,0} + kg_i. \tag{3.37}$$

This is the difference equation used for the first row. Thus, to implement the explicit finite difference algorithm, we use Eq. 3.37 for the first time step and Eq. 3.31 for all subsequent time steps.

3.2.4 Nonreflecting Boundaries

In some applications, it is of interest to model domains that are large enough to be considered infinite in extent. In a finite difference representation of the domain, an infinite boundary has

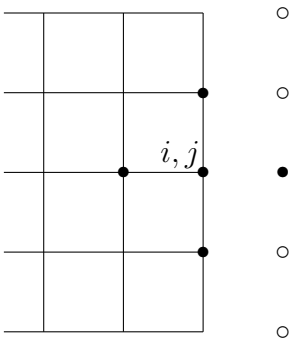


Figure 21: Finite Difference Mesh at Nonreflecting Boundary.

to be truncated at some sufficiently large distance. At such a boundary, a suitable boundary condition must be imposed to ensure that outgoing waves are not reflected.

Consider a vibrating string which extends to infinity for large x. We truncate the computational domain at some finite x. Let the initial velocity be zero. The d'Alembert solution, Eq. 3.20, of the one-dimensional wave equation $c^2 u_{xx} = u_{tt}$ can be written in the form

$$u(x,t) = F_1(x - ct) + F_2(x + ct), \tag{3.38}$$

where F_1 represents a wave advancing at speed c toward the boundary, and F_2 represents the returning wave, which should not exist if the boundary is nonreflecting. With $F_2 = 0$, we differentiate u with respect to x and t to obtain

$$\frac{\partial u}{\partial x} = F_1', \quad \frac{\partial u}{\partial t} = -cF_1', \tag{3.39}$$

where the prime denotes the derivative with respect to the argument. Thus,

$$\frac{\partial u}{\partial x} = -\frac{1}{c}\frac{\partial u}{\partial t}. \tag{3.40}$$

This is the one-dimensional nonreflecting boundary condition. Note that the x direction is normal to the boundary. The boundary condition, Eq. 3.40, must be imposed to inhibit reflections from the truncated boundary. This condition is exact in 1-D (i.e., plane waves) and approximate in higher dimensions, where the nonreflecting condition is written

$$c\,\frac{\partial u}{\partial n} + \frac{\partial u}{\partial t} = 0, \tag{3.41}$$

where n is the outward unit normal to the boundary.

The nonreflecting boundary condition, Eq. 3.40, can be approximated in the finite difference method with central differences expressed in terms of the phantom point outside the boundary. For example, at the typical point (i, j) on the nonreflecting boundary in Fig. 21, the general recursive formula is given by Eq. 3.31:

$$u_{i,j+1} = r^2 u_{i-1,j} + 2(1 - r^2)u_{i,j} + r^2 u_{i+1,j} - u_{i,j-1}. \tag{3.42}$$

The central difference approximation to the nonreflecting boundary condition, Eq. 3.40, is

$$c\,\frac{u_{i+1,j} - u_{i-1,j}}{2h} = -\frac{u_{i,j+1} - u_{i,j-1}}{2k} \tag{3.43}$$

or

$$r\left(u_{i+1,j} - u_{i-1,j}\right) = -\left(u_{i,j+1} - u_{i,j-1}\right). \tag{3.44}$$

The substitution of Eq. 3.44 into Eq. 3.42 (to eliminate the phantom point) yields

$$(1+r)u_{i,j+1} = 2r^2 u_{i-1,j} + 2(1-r^2)u_{i,j} - (1-r)u_{i,j-1}. \tag{3.45}$$

For the first time step ($j = 0$), the last term in this relation is evaluated using the central difference approximation to the initial velocity, Eq. 3.35. Note also that, for $r = 1$, Eq. 3.45 takes a particularly simple (and perhaps unexpected) form:

$$u_{i,j+1} = u_{i-1,j}. \tag{3.46}$$

To illustrate the perfect wave absorption that occurs in a one-dimensional finite difference model, consider an infinitely-long vibrating string with a nonzero initial displacement and zero initial velocity. The initial displacement is a triangular-shaped pulse in the middle of the string, similar to Fig. 14. According to the d'Alembert solution, half the pulse should propagate at speed c to the left and right and be absorbed into the boundaries. We solve the problem with the explicit central finite difference approach with $r = ck/h = 1$. With $r = 1$, the finite difference formulas 3.31 and 3.37 simplify to

$$u_{i,j+1} = u_{i-1,j} + u_{i+1,j} - u_{i,j-1} \tag{3.47}$$

and

$$u_{i,1} = (u_{i-1,0} + u_{i+1,0})/2. \tag{3.48}$$

On the right and left nonreflecting boundaries, Eq. 3.46 implies

$$u_{n,j+1} = u_{n-1,j}, \quad u_{0,j+1} = u_{1,j}, \tag{3.49}$$

where the mesh points in the x direction are labeled 0 to n. The finite difference calculation for this problem results in the following spreadsheet:

t	x=0	x=1	x=2	x=3	x=4	x=5	x=6	x=7	x=8	x=9	x=10
0	0.000	0.000	0.000	1.000	2.000	3.000	2.000	1.000	0.000	0.000	0.000
1	0.000	0.000	0.500	1.000	2.000	2.000	2.000	1.000	0.500	0.000	0.000
2	0.000	0.500	1.000	1.500	1.000	1.000	1.000	1.500	1.000	0.500	0.000
3	0.500	1.000	1.500	1.000	0.500	0.000	0.500	1.000	1.500	1.000	0.500
4	1.000	1.500	1.000	0.500	0.000	0.000	0.000	0.500	1.000	1.500	1.000
5	1.500	1.000	0.500	0.000	0.000	0.000	0.000	0.000	0.500	1.000	1.500
6	1.000	0.500	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.500	1.000
7	0.500	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.500
8	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
9	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
10	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

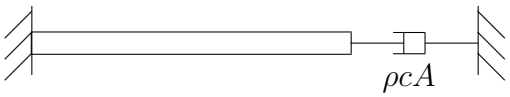


Figure 22: Finite Length Simulation of an Infinite Bar.

Notice that the triangular wave is absorbed without any reflection from the two boundaries.

For steady-state wave motion, the solution $u(x,t)$ is time-harmonic, i.e.,

$$u = u_0 e^{i\omega t}, \tag{3.50}$$

and the nonreflecting boundary condition, Eq. 3.40, becomes

$$\frac{\partial u_0}{\partial x} = -\frac{i\omega}{c} u_0 \tag{3.51}$$

or, in general,

$$\frac{\partial u_0}{\partial n} = -\frac{i\omega}{c} u_0 = -iku_0, \tag{3.52}$$

where n is the outward unit normal at the nonreflecting boundary, and $k = \omega/c$ is the wave number. This condition is exact in 1-D (i.e., plane waves) and approximate in higher dimensions.

The nonreflecting boundary condition can be interpreted physically as a damper (dashpot). Consider, for example, a bar undergoing longitudinal vibration and terminated on the right end with the nonreflecting boundary condition, Eq. 3.40 (Fig. 22). The internal longitudinal force F in the bar is given by

$$F = A\sigma = AE\varepsilon = AE\frac{\partial u}{\partial x}, \tag{3.53}$$

where A is the cross-sectional area of the bar, σ is the stress, E is the Young's modulus of the bar material, and u is displacement. Thus, from Eq. 3.40, the nonreflecting boundary condition is equivalent to applying an end force given by

$$F = -\frac{AE}{c} v, \tag{3.54}$$

where $v = \partial u/\partial t$ is the velocity. Since, from Eq. 2.10,

$$E = \rho c^2, \tag{3.55}$$

Eq. 3.54 becomes

$$F = -(\rho cA)v, \tag{3.56}$$

which is a force proportional to velocity. A mechanical device which applies a force proportional to velocity is a dashpot. The minus sign in this equation means that the force opposes the direction of motion, as required to be physically realizable. The dashpot constant is ρcA. Thus, the application of this dashpot to the end of a finite length bar simulates exactly a bar of infinite length (Fig. 22). Since, in acoustics, the ratio of pressure (force/area) to velocity is referred to as impedance, we see that the characteristic impedance of an acoustic medium is ρc.

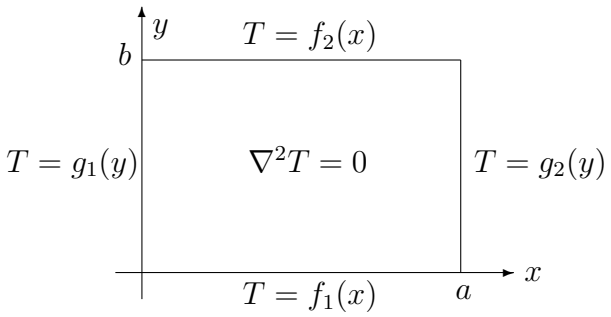


Figure 23: Laplace's Equation on Rectangular Domain.

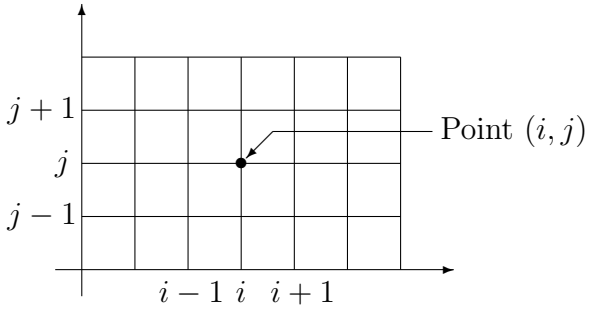


Figure 24: Finite Difference Grid on Rectangular Domain.

3.3 Elliptic Equations

Consider Laplace's equation on the two-dimensional rectangular domain shown in Fig. 23:

$$\begin{cases} \nabla^2 T(x,y) = 0 \quad (0 < x < a,\ 0 < y < b), \\ T(0,y) = g_1(y), \quad T(a,y) = g_2(y), \quad T(x,0) = f_1(x), \quad T(x,b) = f_2(x). \end{cases} \tag{3.57}$$

This problem corresponds physically to two-dimensional steady-state heat conduction over a rectangular plate for which temperature is specified on the boundary.

We attempt an approximate solution by introducing a uniform rectangular grid over the domain, and let the point (i, j) denote the point having the ith value of x and the jth value of y (Fig. 24). Then, using central finite difference approximations to the second derivatives (Fig. 25),

$$\frac{\partial^2 T}{\partial x^2} \approx \frac{T_{i-1,j} - 2T_{i,j} + T_{i+1,j}}{h^2}, \tag{3.58}$$

$$\frac{\partial^2 T}{\partial y^2} \approx \frac{T_{i,j-1} - 2T_{i,j} + T_{i,j+1}}{h^2}. \tag{3.59}$$

The finite difference approximation to Laplace's equation thus becomes

$$\frac{T_{i-1,j} - 2T_{i,j} + T_{i+1,j}}{h^2} + \frac{T_{i,j-1} - 2T_{i,j} + T_{i,j+1}}{h^2} = 0 \tag{3.60}$$

or

$$4T_{i,j} - (T_{i-1,j} + T_{i+1,j} + T_{i,j-1} + T_{i,j+1}) = 0. \tag{3.61}$$

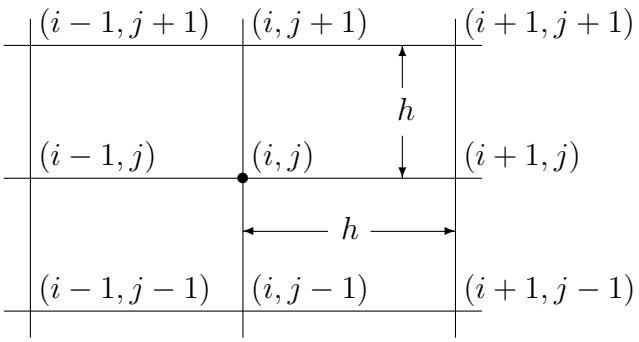


Figure 25: The Neighborhood of Point (i, j).

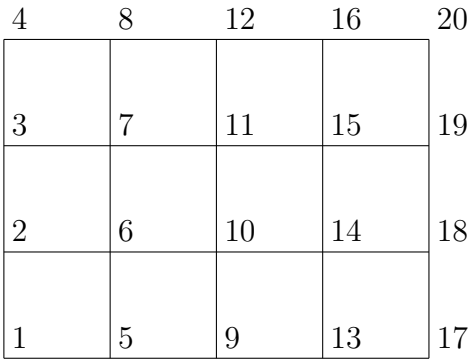


Figure 26: 20-Point Finite Difference Mesh.

That is, for Laplace's equation with the same uniform mesh in each direction, the solution at a typical point (i, j) is the *average* of the four neighboring points.

For example, consider the mesh shown in Fig. 26. Although there are 20 mesh points, 14 are on the boundary, where the temperature is known. Thus, the resulting numerical problem has only six degrees of freedom (unknown variables). The application of Eq. 3.61 to each of the six interior points yields

$$\left\{\begin{array}{rcrcrcrcrcrcl} 4T_6 & - & T_7 & - & T_{10} & & & & & & & = & T_2 + T_5 \\ -T_6 & + & 4T_7 & & & - & T_{11} & & & & & = & T_3 + T_8 \\ -T_6 & & & + & 4T_{10} & - & T_{11} & - & T_{14} & & & = & T_9 \\ & & -T_7 & - & T_{10} & + & 4T_{11} & & & - & T_{15} & = & T_{12} \\ & & & & -T_{10} & & & + & 4T_{14} & - & T_{15} & = & T_{13} + T_{18} \\ & & & & & & -T_{11} & - & T_{14} & + & 4T_{15} & = & T_{16} + T_{19}, \end{array}\right. \tag{3.62}$$

where all known quantities have been placed on the right-hand side. This linear system of six equations in six unknowns can be solved with standard equation solvers. Because the central difference operator is a 5-point operator, systems of equations of this type would have at most five nonzero terms in each equation, regardless of how large the mesh is. Thus, for large meshes, the system of equations is sparsely populated, so that sparse matrix solution techniques would be applicable.

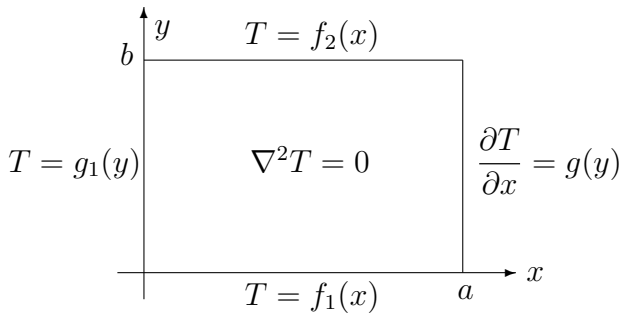


Figure 27: Laplace's Equation With Dirichlet and Neumann B.C.

Since the numbers assigned to the mesh points in Fig. 26 are merely labels for identification, the pattern of nonzero coefficients appearing on the left-hand side of Eq. 3.62 depends on the choice of mesh ordering. Some equation solvers based on Gaussian elimination operate more efficiently on systems of equations for which the nonzeros in the coefficient matrix are clustered near the main diagonal. Such a matrix system is called *banded*.

Systems of this type can also be solved using an iterative procedure known as *relaxation*, which uses the following general algorithm:

1. Initialize the boundary points to their prescribed values, and initialize the interior points to zero or some other convenient value (e.g., the average of the boundary values).

2. Loop systematically through the interior mesh points, setting each interior point to the average of its four neighbors.

3. Continue this process until the solution converges to the desired accuracy.

3.3.1 Derivative Boundary Conditions

The approach of the preceding section must be modified for Neumann boundary conditions, in which the normal derivative is specified. For example, consider again the problem of the last section but with a Neumann, rather than Dirichlet, boundary condition on the right side (Fig. 27):

$$\begin{cases} \nabla^2 T(x,y) = 0 \quad (0 < x < a,\ 0 < y < b), \\ T(0,y) = g_1(y), \quad \dfrac{\partial T(a,y)}{\partial x} = g(y), \quad T(x,0) = f_1(x), \quad T(x,b) = f_2(x). \end{cases} \tag{3.63}$$

We extend the mesh to include additional points to the right of the boundary at $x = a$ (Fig. 28). At a typical point on the boundary, a central difference approximation yields

$$g_{18} = \frac{\partial T_{18}}{\partial x} \approx \frac{T_{22} - T_{14}}{2h} \tag{3.64}$$

or

$$T_{22} = T_{14} + 2hg_{18}. \tag{3.65}$$

On the other hand, the equilibrium equation for Point 18 is

$$4T_{18} - (T_{14} + T_{22} + T_{17} + T_{19}) = 0, \tag{3.66}$$

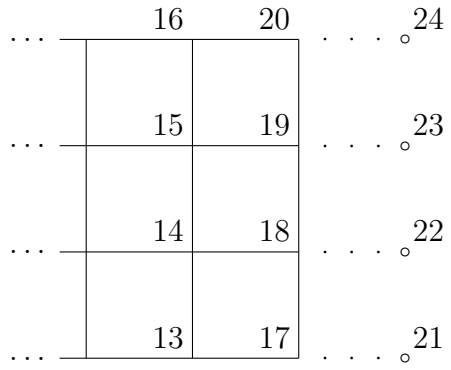


Figure 28: Treatment of Neumann Boundary Conditions.

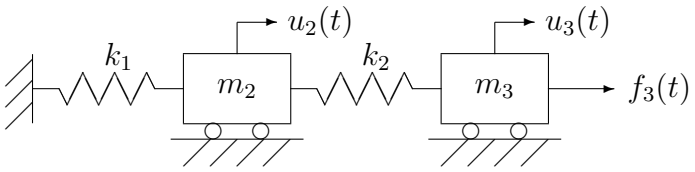


Figure 29: 2-DOF Mass-Spring System.

which, when combined with Eq. 3.65, yields

$$4T_{18} - 2T_{14} - T_{17} - T_{19} = 2hg_{18}, \tag{3.67}$$

which imposes the Neumann boundary condition on Point 18.

4 Direct Finite Element Analysis

The finite element method is a numerical procedure for solving partial differential equations. The procedure is used in a variety of applications, including structural mechanics and dynamics, acoustics, heat transfer, fluid flow, electric and magnetic fields, and electromagnetics. Although the main theoretical bases for the finite element method are variational principles and the weighted residual method, it is useful to consider discrete systems first to gain some physical insight into some of the procedures.

4.1 Linear Mass-Spring Systems

Consider the two-degree-of-freedom (DOF) system shown in Fig. 29. We let u_2 and u_3 denote the displacements from the equilibrium of the two masses m_2 and m_3. The stiffnesses of the two springs are k_1 and k_2. The dynamic equilibrium equations could be obtained from Newton's second law (F=ma):

$$\begin{aligned} m_2\ddot{u}_2 + k_1u_2 - k_2(u_3 - u_2) &= 0 \\ m_3\ddot{u}_3 + k_2(u_3 - u_2) &= f_3(t) \end{aligned} \tag{4.1}$$

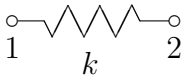


Figure 30: A Single Spring Element.

or

$$\begin{aligned} m_2\ddot{u}_2 + (k_1+k_2)u_2 - k_2u_3 &= 0 \\ m_3\ddot{u}_3 - k_2u_2 + k_2u_3 &= f_3(t). \end{aligned} \tag{4.2}$$

This system could be rewritten in matrix notation as

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{F}(t), \tag{4.3}$$

where

$$\mathbf{u} = \left\{ \begin{array}{c} u_2 \\ u_3 \end{array} \right\} \tag{4.4}$$

is the displacement vector (the vector of unknown displacements),

$$\mathbf{K} = \left[\begin{array}{cc} k_1+k_2 & -k_2 \\ -k_2 & k_2 \end{array} \right] \tag{4.5}$$

is the system stiffness matrix,

$$\mathbf{M} = \left[\begin{array}{cc} m_2 & 0 \\ 0 & m_3 \end{array} \right] \tag{4.6}$$

is the system mass matrix, and

$$\mathbf{F} = \left\{ \begin{array}{c} 0 \\ f_3 \end{array} \right\} \tag{4.7}$$

is the force vector. This approach would be very tedious and error-prone for more complex systems involving many springs and masses.

To develop instead a matrix approach, we first isolate one element, as shown in Fig. 30. The stiffness matrix $\mathbf{K}^{\text{el}}$ for this element satisfies

$$\mathbf{K}^{\text{el}}\mathbf{u} = \mathbf{F}, \tag{4.8}$$

or

$$\left[\begin{array}{cc} k_{11} & k_{12} \\ k_{21} & k_{22} \end{array} \right] \left\{ \begin{array}{c} u_1 \\ u_2 \end{array} \right\} = \left\{ \begin{array}{c} f_1 \\ f_2 \end{array} \right\}. \tag{4.9}$$

By expanding this equation, we obtain

$$\left. \begin{array}{l} k_{11}u_1 + k_{12}u_2 = f_1 \\ k_{21}u_1 + k_{22}u_2 = f_2 \end{array} \right\}. \tag{4.10}$$

From this equation, we observe that k_{11} can be defined as the force on DOF 1 corresponding to enforcing a unit displacement on DOF 1 and zero displacement on DOF 2:

$$k_{11} = f_1\big|_{u_1=1,u_2=0} = k. \tag{4.11}$$

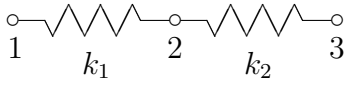


Figure 31: 3-DOF Spring System.

Similarly,

$$k_{12} = f_1\big|_{u_1=0,u_2=1} = -k, \quad k_{21} = f_2\big|_{u_1=1,u_2=0} = -k, \quad k_{22} = f_2\big|_{u_1=0,u_2=1} = k. \tag{4.12}$$

Thus,

$$\mathbf{K}^{\text{el}} = \begin{bmatrix} k & -k \\ -k & k \end{bmatrix} = k \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}. \tag{4.13}$$

In general, for a larger system with many more DOF,

$$\left.\begin{aligned} K_{11}u_1 + K_{12}u_2 + K_{13}u_3 + \cdots + K_{1n}u_n &= F_1 \\ K_{21}u_1 + K_{22}u_2 + K_{23}u_3 + \cdots + K_{2n}u_n &= F_2 \\ K_{31}u_1 + K_{32}u_2 + K_{33}u_3 + \cdots + K_{3n}u_n &= F_3 \\ \vdots & \\ K_{n1}u_1 + K_{n2}u_2 + K_{n3}u_3 + \cdots + K_{nn}u_n &= F_n \end{aligned}\right\}, \tag{4.14}$$

in which case we can interpret an individual element K_{ij} in the stiffness matrix as the force at DOF i if $u_j = 1$ and all other displacement components $u_i = 0$:

$$K_{ij} = F_i\big|_{u_j=1,\ \text{others}=0}. \tag{4.15}$$

4.2 Matrix Assembly

We now return to the stiffness part of the original problem shown in Fig. 29. In the absence of the masses and constraints, this system is shown in Fig. 31. Since there are three points in this system, each with one DOF, the system is a 3-DOF system. The system stiffness matrix can be assembled for this system by adding the 3×3 stiffness matrices for each element:

$$\mathbf{K} = \begin{bmatrix} k_1 & -k_1 & 0 \\ -k_1 & k_1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & k_2 & -k_2 \\ 0 & -k_2 & k_2 \end{bmatrix} = \begin{bmatrix} k_1 & -k_1 & 0 \\ -k_1 & k_1 + k_2 & -k_2 \\ 0 & -k_2 & k_2 \end{bmatrix}. \tag{4.16}$$

The justification for this assembly procedure is that forces are additive. For example, k_{22} is the force at DOF 2 when $u_2 = 1$ and $u_1 = u_3 = 0$; both elements which connect to DOF 2 contribute to k_{22}. This matrix corresponds to the unconstrained system.

4.3 Constraints

The system in Fig. 29 has a constraint on DOF 1, as shown in Fig. 32. If we expand the

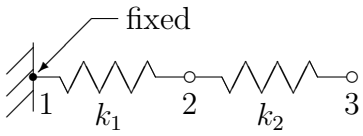


Figure 32: Spring System With Constraint.

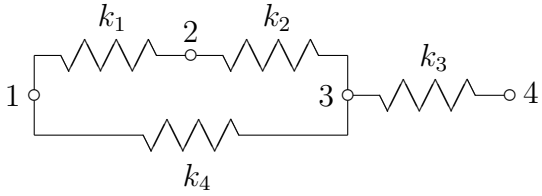


Figure 33: 4-DOF Spring System.

(unconstrained) matrix system $\mathbf{Ku} = \mathbf{F}$ into

$$\left.\begin{aligned} K_{11}u_1 + K_{12}u_2 + K_{13}u_3 &= F_1 \\ K_{21}u_1 + K_{22}u_2 + K_{23}u_3 &= F_2 \\ K_{31}u_1 + K_{32}u_2 + K_{33}u_3 &= F_3 \end{aligned}\right\}, \tag{4.17}$$

we see that Row i corresponds to the equilibrium equation for DOF i. Thus, if $u_i = 0$, we do not need Row i of the matrix (although that equation can be saved to recover later the constraint force). Also, Column i of the matrix multiplies u_i, so that, if $u_i = 0$, we do not need Column i of the matrix. That is, if $u_i = 0$ for some system, we can enforce that constraint by deleting Row i and Column i from the unconstrained matrix. Hence, with $u_1 = 0$ in Fig. 32, we delete Row 1 and Column 1 in Eq. 4.16 to obtain the reduced matrix

$$\mathbf{K} = \begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix}, \tag{4.18}$$

which is the same matrix obtained previously in Eq. 4.5.

Notice that, from Eqs. 4.16 and 4.18,

$$\det \mathbf{K}_{3\times 3} = k_1[(k_1 + k_2)k_2 - k_2^2] + k_1(-k_1 k_2) = 0, \tag{4.19}$$

whereas

$$\det \mathbf{K}_{2\times 2} = (k_1 + k_2)k_2 - k_2^2 = k_1 k_2 \neq 0. \tag{4.20}$$

That is, the unconstrained matrix $\mathbf{K}_{3\times 3}$ is singular, but the constrained matrix $\mathbf{K}_{2\times 2}$ is nonsingular. Without constraints, $\mathbf{K}$ is singular (and the solution of the mechanical problem is not unique) because of the presence of rigid body modes.

4.4 Example and Summary

Consider the 4-DOF spring system shown in Fig. 33. The unconstrained stiffness matrix for

this system is

$$\mathbf{K} = \begin{bmatrix} k_1 + k_4 & -k_1 & -k_4 & 0 \\ -k_1 & k_1 + k_2 & -k_2 & 0 \\ -k_4 & -k_2 & k_2 + k_3 + k_4 & -k_3 \\ 0 & 0 & -k_3 & k_3 \end{bmatrix}. \tag{4.21}$$

We summarize several properties of stiffness matrices:

1. $\mathbf{K}$ is symmetric. This property is a special case of the Betti reciprocal theorem in mechanics.

2. An off-diagonal term is zero unless the two points are common to the same element. Thus, $\mathbf{K}$ is sparse in general and usually banded.

3. $\mathbf{K}$ is singular without enough constraints to eliminate rigid body motion.

For spring systems, that have only one DOF at each point, the sum of any matrix column or row is zero. This property is a consequence of equilibrium, since the matrix entries in Column i consist of all the grid point forces when $u_i = 1$ and other DOF are fixed. The forces must sum to zero, since the object is in static equilibrium.

We summarize the solution procedure for spring systems:

1. Generate the element stiffness matrices.

2. Assemble the system $\mathbf{K}$ and $\mathbf{F}$.

3. Apply constraints.

4. Solve $\mathbf{Ku} = \mathbf{F}$ for $\mathbf{u}$.

5. Compute reactions, spring forces, and stresses.

4.5 Pin-Jointed Rod Element

Consider the pin-jointed rod element (an axial member) shown in Fig. 34. From mechanics of materials, we recall that the change in displacement u for a rod of length L subjected to an axial force F is

$$u = \frac{FL}{AE}, \tag{4.22}$$

where A is the rod cross-sectional area, and E is the Young's modulus for the rod material. Thus, the axial stiffness is

$$k = \frac{F}{u} = \frac{AE}{L}. \tag{4.23}$$

The rod is therefore equivalent to a scalar spring with $k = AE/L$, and

$$\mathbf{K}^{\text{el}} = \frac{AE}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}. \tag{4.24}$$

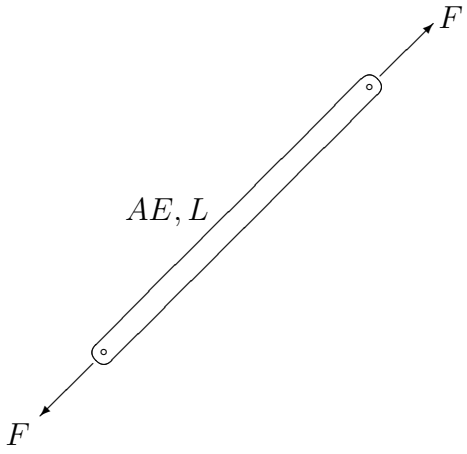


Figure 34: Pin-Jointed Rod Element.

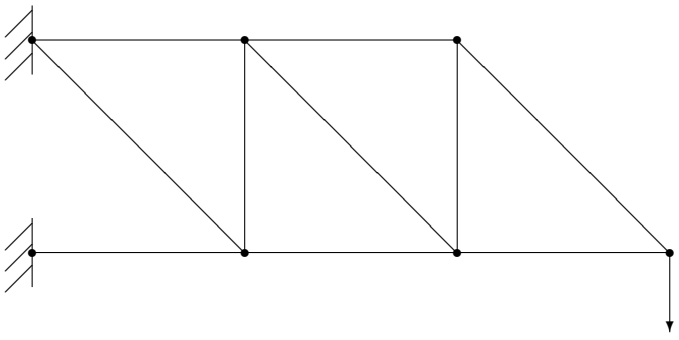


Figure 35: Truss Structure Modeled With Pin-Jointed Rods.

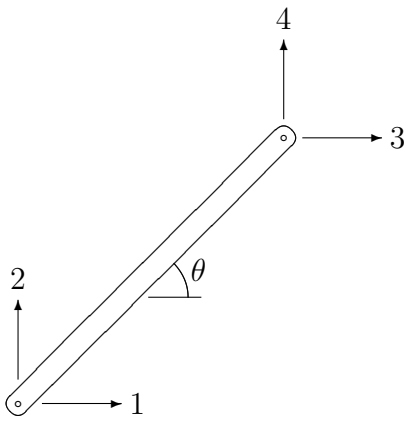


Figure 36: The Degrees of Freedom for a Pin-Jointed Rod Element in 2-D.

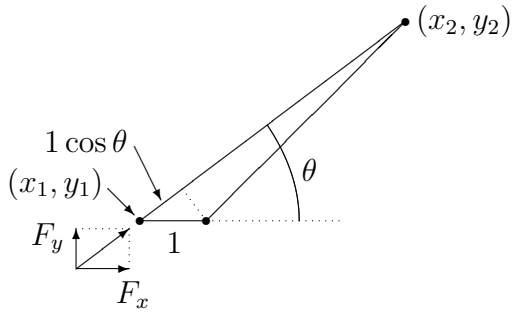


Figure 37: Computing 2-D Stiffness of Pin-Jointed Rod.

However, matrix assembly for a truss structure (a structure made of pin-jointed rods) differs from that for a collection of springs, since the rod elements are not all colinear (e.g., Fig. 35). Thus the elements must be transformed to a common coordinate system before the element matrices can be assembled into the system stiffness matrix.

A typical rod element in 2-D is shown in Fig. 36. To use this element in 2-D trusses requires expanding the 2×2 matrix $\mathbf{K}^{\text{el}}$ into a 4×4 matrix. The four DOF for the element are also shown in Fig. 36. To compute the entries in the first column of the 4×4 $\mathbf{K}^{\text{el}}$, we set $u_1 = 1$, $u_2 = u_3 = u_4 = 0$, and compute the four grid point forces F_1, F_2, F_3, F_4, as shown in Fig. 37. For example, at Point 1,

$$F_x = F\cos\theta = (k \cdot 1\cos\theta)\cos\theta = k\cos^2\theta = k_{11} = -k_{31} \tag{4.25}$$

$$F_y = F\sin\theta = (k \cdot 1\cos\theta)\sin\theta = k\cos\theta\sin\theta = k_{21} = -k_{41}, \tag{4.26}$$

where $k = AE/L$, and the forces at the opposite end of the rod (x_2, y_2) are equal in magnitude and opposite in sign. These four values complete the first column of the matrix. Similarly we can find the rest of the matrix to obtain

$$\mathbf{K}^{\text{el}} = \frac{AE}{L}\begin{bmatrix} c^2 & cs & -c^2 & -cs \\ cs & s^2 & -cs & -s^2 \\ -c^2 & -cs & c^2 & cs \\ -cs & -s^2 & cs & s^2 \end{bmatrix}, \tag{4.27}$$

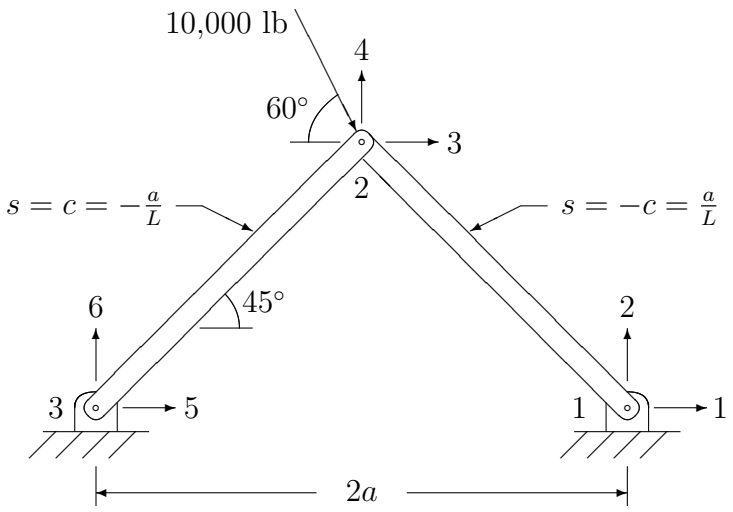


Figure 38: Pin-Jointed Frame Example.

where

$$c = \cos\theta = \frac{x_2 - x_1}{L}, \quad s = \sin\theta = \frac{y_2 - y_1}{L}. \tag{4.28}$$

Note that the angle θ is not of interest; only c and s are needed. Later, in the discussion of matrix transformations, we will derive a more convenient way to obtain this matrix.

4.6 Pin-Jointed Frame Example

We illustrate the matrix assembly and solution procedures for truss structures with the simple frame shown in Fig. 38. For this frame, we assume $E = 30 \times 10^6$ psi, L=10 in, and A=1 in^2. Before the application of constraints, the stiffness matrix is

$$\mathbf{K} = k_0 \begin{bmatrix} 1 & -1 & -1 & 1 & 0 & 0 \\ -1 & 1 & 1 & -1 & 0 & 0 \\ -1 & 1 & 1+1 & -1+1 & -1 & -1 \\ 1 & -1 & -1+1 & 1+1 & -1 & -1 \\ 0 & 0 & -1 & -1 & 1 & 1 \\ 0 & 0 & -1 & -1 & 1 & 1 \end{bmatrix}, \tag{4.29}$$

where

$$k_0 = \frac{AE}{L}\left(\frac{a}{L}\right)^2 = 1.5 \times 10^6 \text{ lb/in}, \quad \frac{a}{L} = \frac{1}{\sqrt{2}}. \tag{4.30}$$

After constraints, the system of equations is

$$k_0 \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{Bmatrix} u_3 \\ u_4 \end{Bmatrix} = \begin{Bmatrix} 5000 \\ -5000\sqrt{3} \end{Bmatrix}, \tag{4.31}$$

whose solution is

$$u_3 = 1.67 \times 10^{-3} \text{ in}, \quad u_4 = -2.89 \times 10^{-3} \text{ in}. \tag{4.32}$$

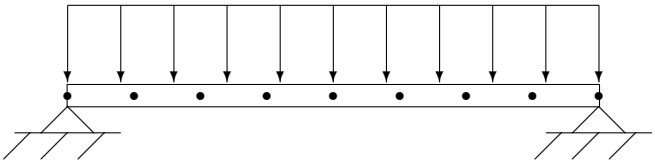


Figure 39: Example With Reactions and Loads at Same DOF.

The reactions can be recovered as

$$\begin{aligned} R_1 &= k_0(-u_3 + u_4) = -6840 \text{ lb}, & R_2 &= k_0(u_3 - u_4) = 6840 \text{ lb} \\ R_5 &= k_0(-u_3 - u_4) = 1830 \text{ lb}, & R_6 &= k_0(-u_3 - u_4) = 1830 \text{ lb}. \end{aligned} \tag{4.33}$$

4.7 Boundary Conditions by Matrix Partitioning

We recall that, to enforce $u_i = 0$, we could delete Row i and Column i from the stiffness matrix. By using matrix partitioning, we can treat nonzero constraints and recover the forces of constraint.

Consider the finite element matrix system $\mathbf{Ku} = \mathbf{F}$, where some DOF are specified, and some are free. We partition the unknown displacement vector into

$$\mathbf{u} = \left\{ \begin{array}{c} \mathbf{u}_f \\ \mathbf{u}_s \end{array} \right\}, \tag{4.34}$$

where $\mathbf{u}_f$ and $\mathbf{u}_s$ denote, respectively, the free and constrained DOF. A partitioning of the matrix system then yields

$$\left[\begin{array}{cc} \mathbf{K}_{ff} & \mathbf{K}_{fs} \\ \mathbf{K}_{sf} & \mathbf{K}_{ss} \end{array} \right] \left\{ \begin{array}{c} \mathbf{u}_f \\ \mathbf{u}_s \end{array} \right\} = \left\{ \begin{array}{c} \mathbf{F}_f \\ \mathbf{F}_s \end{array} \right\}, \tag{4.35}$$

where $\mathbf{u}_s$ is prescribed. From the first partition,

$$\mathbf{K}_{ff}\mathbf{u}_f + \mathbf{K}_{fs}\mathbf{u}_s = \mathbf{F}_f \tag{4.36}$$

or

$$\mathbf{K}_{ff}\mathbf{u}_f = \mathbf{F}_f - \mathbf{K}_{fs}\mathbf{u}_s, \tag{4.37}$$

which can be solved for the unknown $\mathbf{u}_f$. The second partition then yields the forces at the constrained DOF:

$$\mathbf{F}_s = \mathbf{K}_{sf}\mathbf{u}_f + \mathbf{K}_{ss}\mathbf{u}_s, \tag{4.38}$$

where $\mathbf{u}_f$ is now known, and $\mathbf{u}_s$ is prescribed.

The grid point forces $\mathbf{F}_s$ at the constrained DOF consist of both the reactions of constraint and the applied loads, if any, at those DOF. For example, consider the beam shown in Fig. 39. When the applied load is distributed to the grid points, the loads at the two end grid points would include both reactions and a portion of the applied load. Thus, $\mathbf{F}_s$, which includes all loads at the constrained DOF, can be written as

$$\mathbf{F}_s = \mathbf{P}_s + \mathbf{R}, \tag{4.39}$$

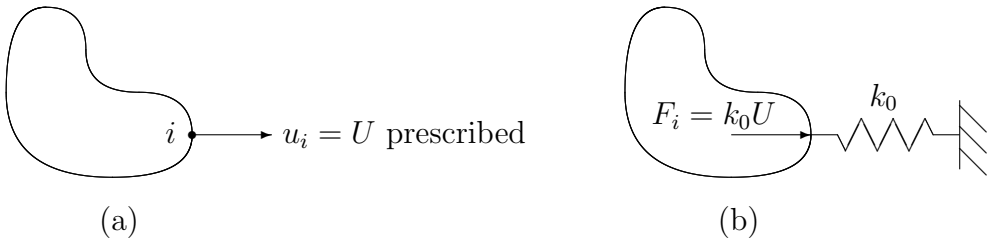


Figure 40: Large Spring Approach to Constraints.

where $\mathbf{P}_s$ is the applied load, and $\mathbf{R}$ is the vector of reactions. The reactions can then be recovered as

$$\mathbf{R} = \mathbf{K}_{sf}\mathbf{u}_f + \mathbf{K}_{ss}\mathbf{u}_s - \mathbf{P}_s. \tag{4.40}$$

The disadvantage of using the partitioning approach to constraints is that the writing of computer programs is made more complicated, since one needs the general capability to partition a matrix into smaller matrices. However, for general purpose software, such a capability is essential.

4.8 Alternative Approach to Constraints

Consider a structure (Fig. 40) for which we want to prescribe a value for the ith DOF: $u_i = U$. An alternative approach to enforce this constraint is to attach a large spring k_0 between DOF i and ground (a fixed point) and to apply a force F_i to DOF i equal to $k_0 U$. This spring should be many orders of magnitude larger than other typical values in the stiffness matrix. A large number placed on the matrix diagonal will have no adverse effects on the conditioning of the matrix.

For example, if we prescribe DOF 3 in a 4-DOF system, the modified system becomes

$$\begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} \\ k_{21} & k_{22} & k_{23} & k_{24} \\ k_{31} & k_{32} & k_{33}+k_0 & k_{34} \\ k_{41} & k_{42} & k_{43} & k_{44} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{Bmatrix} = \begin{Bmatrix} F_1 \\ F_2 \\ F_3 + k_0 U \\ F_4 \end{Bmatrix}. \tag{4.41}$$

For large k_0, the third equation is

$$k_0 u_3 \approx k_0 U \quad \text{or} \quad u_3 \approx U, \tag{4.42}$$

as required.

The large spring approach can be used for any system of equations for which one wants to enforce a constraint on a particular unknown. The main advantage of this approach is that computer coding is easier, since matrix sizes do not have to change.

We can summarize the algorithm for applying the large-spring approach for applying constraints to the linear system $\mathbf{Ku} = \mathbf{F}$ as follows:

1. Choose the large spring k_0 to be, say, 10,000 times the largest diagonal entry in the unconstrained $\mathbf{K}$ matrix.

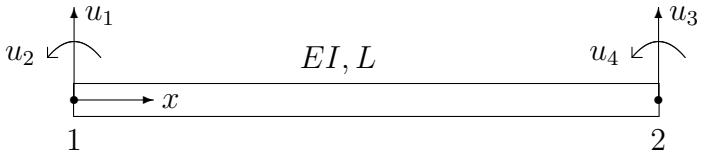


Figure 41: DOF for Beam in Flexure (2-D).

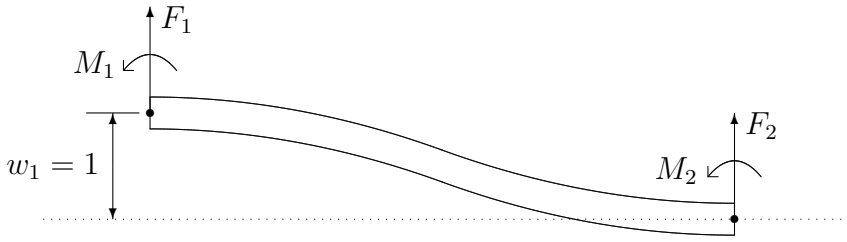


Figure 42: The Beam Problem Associated With Column 1.

2. For each DOF i which is to be constrained (zero or not), add k_0 to the diagonal entry K_{ii}, and add $k_0 U$ to the corresponding right-hand side term F_i, where U is the desired constraint value for DOF i.

4.9 Beams in Flexure

Like springs and pin-jointed rods, beam elements also have stiffness matrices which can be computed exactly. In two dimensions, we designate the four DOF associated with flexure as shown in Fig. 41. The stiffness matrix would therefore be of the form

$$\left\{ \begin{array}{c} F_1 \\ M_1 \\ F_2 \\ M_2 \end{array} \right\} = \left[\begin{array}{cccc} k_{11} & k_{12} & k_{13} & k_{14} \\ k_{21} & k_{22} & k_{23} & k_{24} \\ k_{31} & k_{32} & k_{33} & k_{34} \\ k_{41} & k_{42} & k_{43} & k_{44} \end{array} \right] \left\{ \begin{array}{c} w_1 \\ \theta_1 \\ w_2 \\ \theta_2 \end{array} \right\}, \tag{4.43}$$

where w_i and θ_i are, respectively, the transverse displacement and rotation at the ith point. Rotations are expressed in radians. To compute the first column of $\mathbf{K}$, set

$$w_1 = 1, \quad \theta_1 = w_2 = \theta_2 = 0, \tag{4.44}$$

and compute the four forces. Thus, for an Euler beam, we solve the beam differential equation

$$EIy'' = M(x), \tag{4.45}$$

subject to the boundary conditions, Eq. 4.44, as shown in Fig. 42. For Column 2, we solve the beam equation subject to the boundary conditions

$$\theta_1 = 1, \quad w_1 = w_2 = \theta_2 = 0, \tag{4.46}$$

as shown in Fig. 43. If we then combine the resulting flexural stiffnesses with the axial

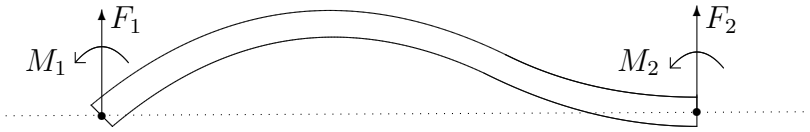


Figure 43: The Beam Problem Associated With Column 2.

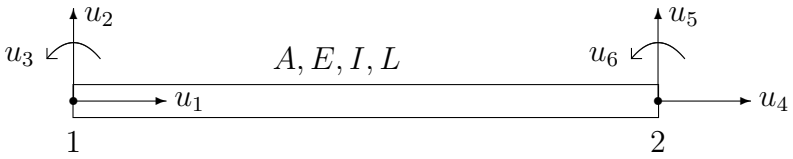


Figure 44: DOF for 2-D Beam Element.

stiffnesses previously derived for the axial member, we obtain the two-dimensional stiffness matrix for the Euler beam:

$$\begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \\ F_6 \end{Bmatrix} = \begin{bmatrix} \frac{AE}{L} & 0 & 0 & -\frac{AE}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{AE}{L} & 0 & 0 & \frac{AE}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{Bmatrix}, \qquad (4.47)$$

where the six DOF are shown in Fig. 44.

For this element, note that there is no coupling between axial and transverse behavior. Transverse shear, which was ignored, can be added. The three-dimensional counterpart to this matrix would have six DOF at each grid point: three translations and three rotations $(u_x, u_y, u_z, R_x, R_y, R_z)$. Thus, in 3-D, $\mathbf{K}$ is a 12×12 matrix. In addition, for bending in two different planes, there would have to be two moments of inertia I_1 and I_2, in addition to a torsional constant J and (possibly) a product of inertia I_{12}. For transverse shear, "area factors" would also be needed to reflect the fact that two different beams with the same cross-sectional area, but different cross-sectional shapes, would have different shear stiffnesses.

4.10 Direct Approach to Continuum Problems

Stiffness matrices for springs, rods, and beams can be derived exactly. For most problems of interest in engineering, exact stiffness matrices cannot be derived. Consider the 2-D problem of computing the displacements and stresses in a thin plate with applied forces and

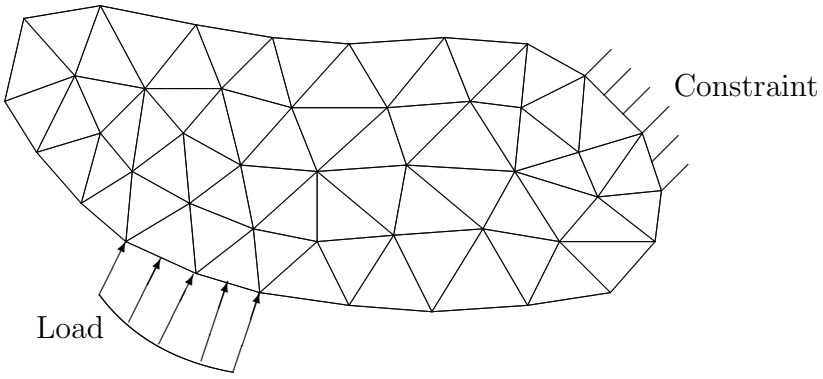


Figure 45: Plate With Constraints and Loads.

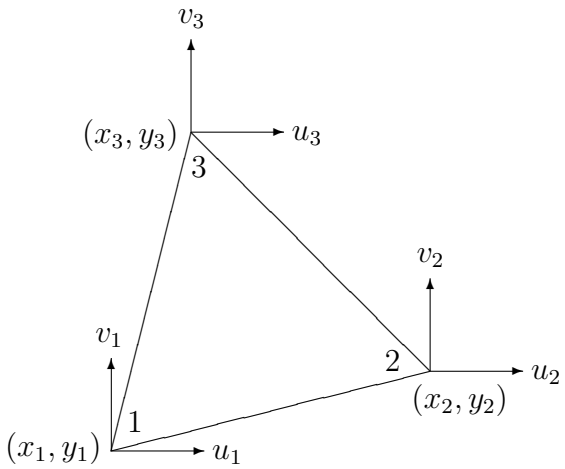


Figure 46: DOF for the Linear Triangular Membrane Element.

constraints (Fig. 45). This figure also shows the domain modeled with several triangular elements. A typical element is shown in Fig. 46. With three grid points and two DOF at each point, this is a 6-DOF element. The displacement and force vectors for the element are

$$\mathbf{u} = \left\{ \begin{array}{c} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \end{array} \right\}, \quad \mathbf{F} = \left\{ \begin{array}{c} F_{1x} \\ F_{1y} \\ F_{2x} \\ F_{2y} \\ F_{3x} \\ F_{3y} \end{array} \right\}. \tag{4.48}$$

Since we cannot determine exactly a stiffness matrix that relates these two vectors, we approximate the displacement field over the element as

$$\left\{ \begin{array}{l} u(x, y) = a_1 + a_2 x + a_3 y \\ v(x, y) = a_4 + a_5 x + a_6 y, \end{array} \right. \tag{4.49}$$

where u and v are the x and y components of displacement, respectively. Note that the number of undetermined coefficients equals the number of DOF (6) in the element. We

choose polynomials for convenience in the subsequent mathematics. The linear approximation in Eq. 4.49 is analogous to the piecewise linear approximation used in trapezoidal rule integration.

At the vertices, the displacements in Eq. 4.49 must match the grid point values, e.g.,

$$u_1 = a_1 + a_2 x_1 + a_3 y_1. \tag{4.50}$$

We can write five more equations of this type to obtain

$$\begin{Bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \end{Bmatrix} = \begin{bmatrix} 1 & x_1 & y_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x_1 & y_1 \\ 1 & x_2 & y_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x_2 & y_2 \\ 1 & x_3 & y_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x_3 & y_3 \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{Bmatrix} \tag{4.51}$$

or

$$\mathbf{u} = \boldsymbol{\gamma}\mathbf{a}. \tag{4.52}$$

Since the x and y directions uncouple in Eq. 4.51, we can write this last equation in uncoupled form:

$$\begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix} = \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \end{Bmatrix} \tag{4.53}$$

$$\begin{Bmatrix} v_1 \\ v_2 \\ v_3 \end{Bmatrix} = \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} \begin{Bmatrix} a_4 \\ a_5 \\ a_6 \end{Bmatrix}. \tag{4.54}$$

We now observe that

$$\det \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} = 2A \neq 0, \tag{4.55}$$

since $|A|$ is (from geometry) the area of the triangle. A is positive for counterclockwise ordering and negative for clockwise ordering. Thus, unless the triangle is degenerate, we conclude that $\boldsymbol{\gamma}$ is invertible, and

$$\begin{Bmatrix} a_1 \\ a_2 \\ a_3 \end{Bmatrix} = \frac{1}{2A} \begin{bmatrix} x_2y_3 - x_3y_2 & x_3y_1 - x_1y_3 & x_1y_2 - x_2y_1 \\ y_2 - y_3 & y_3 - y_1 & y_1 - y_2 \\ x_3 - x_2 & x_1 - x_3 & x_2 - x_1 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix}. \tag{4.56}$$

Similarly,

$$\begin{Bmatrix} a_4 \\ a_5 \\ a_6 \end{Bmatrix} = \frac{1}{2A} \begin{bmatrix} x_2y_3 - x_3y_2 & x_3y_1 - x_1y_3 & x_1y_2 - x_2y_1 \\ y_2 - y_3 & y_3 - y_1 & y_1 - y_2 \\ x_3 - x_2 & x_1 - x_3 & x_2 - x_1 \end{bmatrix} \begin{Bmatrix} v_1 \\ v_2 \\ v_3 \end{Bmatrix}. \tag{4.57}$$

Thus, we can write

$$\mathbf{a} = \boldsymbol{\gamma}^{-1}\mathbf{u}, \tag{4.58}$$

and treat the element's unknowns as being either the physical displacements $\mathbf{u}$ or the non-physical coefficients $\mathbf{a}$.

The strain components of interest in 2-D are

$$\boldsymbol{\varepsilon} = \left\{ \begin{array}{c} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{array} \right\} = \left\{ \begin{array}{c} u_{,x} \\ v_{,y} \\ u_{,y} + v_{,x} \end{array} \right\}. \tag{4.59}$$

From Eq. 4.49, we evaluate the strains for this element as

$$\boldsymbol{\varepsilon} = \left\{ \begin{array}{c} a_2 \\ a_6 \\ a_3 + a_5 \end{array} \right\} = \left[\begin{array}{cccccc} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{array} \right] \left\{ \begin{array}{c} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{array} \right\} = \mathbf{B}\mathbf{a} = \mathbf{B}\boldsymbol{\gamma}^{-1}\mathbf{u} = \mathbf{C}\mathbf{u}, \tag{4.60}$$

where

$$\mathbf{C} = \mathbf{B}\boldsymbol{\gamma}^{-1}. \tag{4.61}$$

Eq. 4.60 is a formula to compute element strains given the grid point displacements. Note that, for this element, the strains are independent of position in the element. Thus, this element is referred to as the Constant Strain Triangle (CST).

From generalized Hooke's law, each stress component is a linear combination of all the strain components:

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon} = \mathbf{D}\mathbf{B}\boldsymbol{\gamma}^{-1}\mathbf{u} = \mathbf{D}\mathbf{C}\mathbf{u}, \tag{4.62}$$

where, for example, for an isotropic material in plane stress,

$$\mathbf{D} = \frac{E}{1-\nu^2} \left[\begin{array}{ccc} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{array} \right], \tag{4.63}$$

where E is Young's modulus, and ν is Poisson's ratio. Eq. 4.62 is a formula to compute element stresses given the grid point displacements. For this element, the stresses are constant (independent of position) in the element.

We now derive the element stiffness matrix using the Principle of Virtual Work. Consider an element in static equilibrium with a set of applied loads $\mathbf{F}$ and displacements $\mathbf{u}$. According to the Principle of Virtual Work, the work done by the applied loads acting through the displacements is equal to the increment in stored strain energy during an arbitrary virtual (imaginary) displacement $\delta\mathbf{u}$:

$$\delta\mathbf{u}^T\mathbf{F} = \int_V \delta\boldsymbol{\varepsilon}^T\boldsymbol{\sigma}\, dV. \tag{4.64}$$

We then substitute Eqs. 4.60 and 4.62 into this equation to obtain

$$\delta\mathbf{u}^T\mathbf{F} = \int_V (\mathbf{C}\,\delta\mathbf{u})^T(\mathbf{D}\mathbf{C}\mathbf{u})\,dV = \delta\mathbf{u}^T\left(\int_V \mathbf{C}^T\mathbf{D}\mathbf{C}\,dV\right)\mathbf{u}, \tag{4.65}$$

where $\delta\mathbf{u}$ and $\mathbf{u}$ can be removed from the integral, since they are grid point quantities independent of position. Since $\delta\mathbf{u}$ is arbitrary, it follows that

$$\mathbf{F} = \left(\int_V \mathbf{C}^T\mathbf{D}\mathbf{C}\,dV\right)\mathbf{u}. \tag{4.66}$$

Therefore, the stiffness matrix is given by

$$\mathbf{K} = \int_V \mathbf{C}^T\mathbf{D}\mathbf{C}\,dV = \int_V \left(\mathbf{B}\boldsymbol{\gamma}^{-1}\right)^T\mathbf{D}\left(\mathbf{B}\boldsymbol{\gamma}^{-1}\right)dV, \tag{4.67}$$

where the integral is over the volume of the element. Note that, since the material property matrix $\mathbf{D}$ is symmetric, $\mathbf{K}$ is symmetric. The substitution of the constant $\mathbf{B}$, $\boldsymbol{\gamma}$, and $\mathbf{D}$ matrices into this equation yields the stiffness matrix for the Constant Strain Triangle:

$$\mathbf{K} = \frac{Et}{4|A|(1-\nu^2)}\times$$

$$\left[\begin{array}{c|c|c|c|c|c}
y_{23}^2+\lambda x_{32}^2 & \mu x_{32}y_{23} & y_{23}y_{31}+\lambda x_{13}x_{32} & \nu x_{13}y_{23}+\lambda x_{32}y_{31} & y_{12}y_{23}+\lambda x_{21}x_{32} & \nu x_{21}y_{23}+\lambda x_{32}y_{12} \\
 & x_{32}^2+\lambda y_{23}^2 & \nu x_{32}y_{31}+\lambda x_{13}y_{23} & x_{32}x_{13}+\lambda y_{31}y_{23} & \nu x_{32}y_{12}+\lambda x_{21}y_{23} & x_{21}x_{32}+\lambda y_{12}y_{23} \\
 & & y_{31}^2+\lambda x_{13}^2 & \mu x_{13}y_{31} & y_{12}y_{31}+\lambda x_{21}x_{13} & \nu x_{21}y_{31}+\lambda x_{13}y_{12} \\
 & & & x_{13}^2+\lambda y_{31}^2 & \nu x_{13}y_{12}+\lambda x_{21}y_{31} & x_{13}x_{21}+\lambda y_{12}y_{31} \\
 & \text{Sym} & & & y_{12}^2+\lambda x_{21}^2 & \mu x_{21}y_{12} \\
 & & & & & x_{21}^2+\lambda y_{12}^2
\end{array}\right], \tag{4.68}$$

where

$$x_{ij} = x_i - x_j, \quad y_{ij} = y_i - y_j, \quad \lambda = \frac{1-\nu}{2}, \quad \mu = \frac{1+\nu}{2}, \tag{4.69}$$

and t is the element thickness.

5 Change of Basis

On many occasions in engineering applications, the need arises to transform vectors and matrices from one coordinate system to another. For example, in the finite element method, it is frequently more convenient to derive element matrices in a local element coordinate system and then transform those matrices to a global system (Fig. 47). Transformations are also needed to transform from other orthogonal coordinate systems (e.g., cylindrical or spherical) to Cartesian coordinates (Fig. 48).

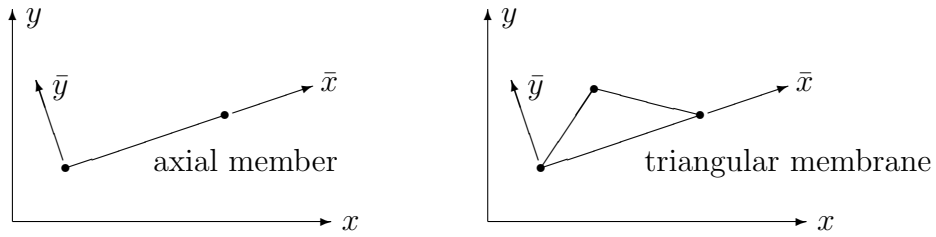


Figure 47: Element Coordinate Systems in the Finite Element Method.

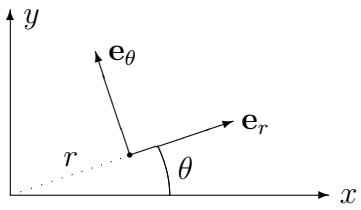


Figure 48: Basis Vectors in Polar Coordinate System.

Let the vector $\mathbf{v}$ be given by

$$\mathbf{v} = v_1\mathbf{e}_1 + v_2\mathbf{e}_2 + v_3\mathbf{e}_3 = \sum_{i=1}^{3} v_i\mathbf{e}_i, \tag{5.1}$$

where $\mathbf{e}_i$ are the basis vectors, and v_i are the components of $\mathbf{v}$. With the *summation convention*, we can omit the summation sign and write

$$\mathbf{v} = v_i\mathbf{e}_i, \tag{5.2}$$

where, if a subscript appears exactly twice, a summation is implied over the range.

An *orthonormal basis* is defined as a basis whose basis vectors are mutually orthogonal unit vectors (i.e., vectors of unit length). If $\mathbf{e}_i$ is an orthonormal basis,

$$\mathbf{e}_i \cdot \mathbf{e}_j = \delta_{ij} = \begin{cases} 1, & i = j, \\ 0, & i \neq j, \end{cases} \tag{5.3}$$

where δ_{ij} is the Kronecker delta.

Since bases are not unique, we can express $\mathbf{v}$ in two different orthonormal bases:

$$\mathbf{v} = \sum_{i=1}^{3} v_i\mathbf{e}_i = \sum_{i=1}^{3} \bar{v}_i\bar{\mathbf{e}}_i, \tag{5.4}$$

where v_i are the components of $\mathbf{v}$ in the unbarred coordinate system, and $\bar{v}_i$ are the components in the barred system (Fig. 49). If we take the dot product of both sides of Eq. 5.4 with $\mathbf{e}_j$, we obtain

$$\sum_{i=1}^{3} v_i\,\mathbf{e}_i \cdot \mathbf{e}_j = \sum_{i=1}^{3} \bar{v}_i\,\bar{\mathbf{e}}_i \cdot \mathbf{e}_j, \tag{5.5}$$

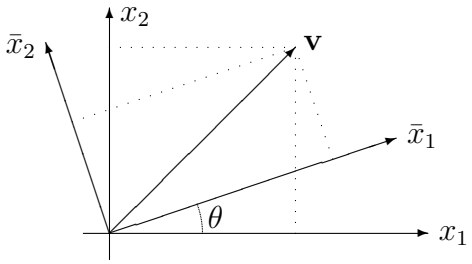


Figure 49: Change of Basis.

where $\mathbf{e}_i \cdot \mathbf{e}_j = \delta_{ij}$, and we define the 3×3 matrix $\mathbf{R}$ as

$$R_{ij} = \bar{\mathbf{e}}_i \cdot \mathbf{e}_j. \tag{5.6}$$

Thus, from Eq. 5.5,

$$v_j = \sum_{i=1}^{3} R_{ij} \bar{v}_i = \sum_{i=1}^{3} R_{ji}^T \bar{v}_i. \tag{5.7}$$

Since the matrix product

$$\mathbf{C} = \mathbf{AB} \tag{5.8}$$

can be written using subscript notation as

$$C_{ij} = \sum_{k=1}^{3} A_{ik} B_{kj}, \tag{5.9}$$

Eq. 5.7 is equivalent to the matrix product

$$\mathbf{v} = \mathbf{R}^T \bar{\mathbf{v}}. \tag{5.10}$$

Similarly, if we take the dot product of Eq. 5.4 with $\bar{\mathbf{e}}_j$, we obtain

$$\sum_{i=1}^{3} v_i \, \mathbf{e}_i \cdot \bar{\mathbf{e}}_j = \sum_{i=1}^{3} \bar{v}_i \, \bar{\mathbf{e}}_i \cdot \bar{\mathbf{e}}_j, \tag{5.11}$$

where $\bar{\mathbf{e}}_i \cdot \bar{\mathbf{e}}_j = \delta_{ij}$, and $\mathbf{e}_i \cdot \bar{\mathbf{e}}_j = R_{ji}$. Thus,

$$\bar{v}_j = \sum_{i=1}^{3} R_{ji} v_i \quad \text{or} \quad \bar{\mathbf{v}} = \mathbf{R}\mathbf{v} \quad \text{or} \quad \mathbf{v} = \mathbf{R}^{-1}\bar{\mathbf{v}}. \tag{5.12}$$

A comparison of Eqs. 5.10 and 5.12 yields

$$\mathbf{R}^{-1} = \mathbf{R}^T \quad \text{or} \quad \mathbf{R}\mathbf{R}^T = \mathbf{I} \quad \text{or} \quad \sum_{k=1}^{3} R_{ik} R_{jk} = \delta_{ij}, \tag{5.13}$$

where $\mathbf{I}$ is the identity matrix ($I_{ij} = \delta_{ij}$):

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{5.14}$$

This type of transformation is called an *orthogonal coordinate transformation* (OCT). A matrix $\mathbf{R}$ satisfying Eq. 5.13 is said to be an *orthogonal* matrix. That is, an orthogonal matrix is one whose inverse is equal to the transpose. $\mathbf{R}$ is sometimes called a *rotation matrix*.

For example, for the coordinate rotation shown in Fig. 49, in 3-D,

$$\mathbf{R} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{5.15}$$

In 2-D,

$$\mathbf{R} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \tag{5.16}$$

and

$$\begin{cases} v_x = \bar{v}_x \cos\theta - \bar{v}_y \sin\theta \\ v_y = \bar{v}_x \sin\theta + \bar{v}_y \cos\theta. \end{cases} \tag{5.17}$$

We recall that the determinant of a matrix product is equal to the product of the determinants. Also, the determinant of the transpose of a matrix is equal to the determinant of the matrix itself. Thus, from Eq. 5.13,

$$\det(\mathbf{R}\mathbf{R}^T) = (\det\mathbf{R})(\det\mathbf{R}^T) = (\det\mathbf{R})^2 = \det\mathbf{I} = 1, \tag{5.18}$$

and we conclude that, for an orthogonal matrix $\mathbf{R}$,

$$\det\mathbf{R} = \pm 1. \tag{5.19}$$

The plus sign occurs for rotations, and the minus sign occurs for combinations of rotations and reflections. For example, the orthogonal matrix

$$\mathbf{R} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{5.20}$$

indicates a reflection in the z direction (i.e., the sign of the z component is changed).

We note that the length of a vector is unchanged under an orthogonal coordinate transformation, since the square of the length is given by

$$\bar{v}_i \bar{v}_i = R_{ij} v_j R_{ik} v_k = \delta_{jk} v_j v_k = v_j v_j, \tag{5.21}$$

where the summation convention was used. That is, the square of the length of a vector is the same in both coordinate systems.

To summarize, under an orthogonal coordinate transformation, vectors transform according to the rule

$$\bar{\mathbf{v}} = \mathbf{R}\mathbf{v} \quad \text{or} \quad \bar{v}_i = \sum_{j=1}^{3} R_{ij} v_j, \tag{5.22}$$

where

$$R_{ij} = \bar{\mathbf{e}}_i \cdot \mathbf{e}_j, \tag{5.23}$$

and

$$\mathbf{R}\mathbf{R}^T = \mathbf{R}^T\mathbf{R} = \mathbf{I}. \tag{5.24}$$

5.1 Tensors

A vector which transforms under an orthogonal coordinate transformation according to the rule $\bar{\mathbf{v}} = \mathbf{R}\mathbf{v}$ is defined as a tensor of rank 1. A tensor of rank 0 is a scalar (a quantity which is unchanged by an orthogonal coordinate transformation). For example, temperature and pressure are scalars, since $\bar{T} = T$ and $\bar{p} = p$.

We now introduce tensors of rank 2. Consider a matrix $\mathbf{M} = (M_{ij})$ which relates two vectors $\mathbf{u}$ and $\mathbf{v}$ by

$$\mathbf{v} = \mathbf{M}\mathbf{u} \quad \text{or} \quad v_i = \sum_{j=1}^{3} M_{ij} u_j \tag{5.25}$$

(i.e., the result of multiplying a matrix and a vector is a vector). Also, in a rotated coordinate system,

$$\bar{\mathbf{v}} = \bar{\mathbf{M}}\bar{\mathbf{u}}. \tag{5.26}$$

Since both $\mathbf{u}$ and $\mathbf{v}$ are vectors (tensors of rank 1), Eq. 5.25 implies

$$\mathbf{R}^T\bar{\mathbf{v}} = \mathbf{M}\mathbf{R}^T\bar{\mathbf{u}} \quad \text{or} \quad \bar{\mathbf{v}} = \mathbf{R}\mathbf{M}\mathbf{R}^T\bar{\mathbf{u}}. \tag{5.27}$$

By comparing Eqs. 5.26 and 5.27, we obtain

$$\bar{\mathbf{M}} = \mathbf{R}\mathbf{M}\mathbf{R}^T \tag{5.28}$$

or, in index notation,

$$\bar{M}_{ij} = \sum_{k=1}^{3}\sum_{l=1}^{3} R_{ik} R_{jl} M_{kl}, \tag{5.29}$$

which is the transformation rule for a tensor of rank 2. In general, a tensor of rank n, which has n indices, transforms under an orthogonal coordinate transformation according to the rule

$$\bar{A}_{ij\cdots k} = \sum_{p=1}^{3}\sum_{q=1}^{3}\cdots\sum_{r=1}^{3} R_{ip} R_{jq} \cdots R_{kr} A_{pq\cdots r}. \tag{5.30}$$

5.2 Examples of Tensors

1. Stress and strain in elasticity

The stress tensor $\boldsymbol{\sigma}$ is

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix}, \tag{5.31}$$

where σ_{11}, σ_{22}, σ_{33} are the direct (normal) stresses, and σ_{12}, σ_{13}, and σ_{23} are the shear stresses. The corresponding strain tensor is

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} & \varepsilon_{13} \\ \varepsilon_{21} & \varepsilon_{22} & \varepsilon_{23} \\ \varepsilon_{31} & \varepsilon_{32} & \varepsilon_{33} \end{bmatrix}, \tag{5.32}$$

where, in terms of displacements,

$$\varepsilon_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i}) = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right). \tag{5.33}$$

The shear strains in this tensor are equal to half the corresponding engineering shear strains. Both $\boldsymbol{\sigma}$ and $\boldsymbol{\varepsilon}$ transform like second rank tensors.

2. Generalized Hooke's law

According to Hooke's law in elasticity, the extension in a bar subjected to an axial force is proportional to the force, or stress is proportional to strain. In 1-D,

$$\sigma = E\varepsilon, \tag{5.34}$$

where E is Young's modulus, an experimentally determined material property.

In general three-dimensional elasticity, there are nine components of stress σ_{ij} and nine components of strain ε_{ij}. According to generalized Hooke's law, each stress component can be written as a linear combination of the nine strain components:

$$\sigma_{ij} = c_{ijkl}\varepsilon_{kl}, \tag{5.35}$$

where the 81 material constants c_{ijkl} are experimentally determined, and the summation convention is being used.

We now prove that c_{ijkl} is a tensor of rank 4. We can write Eq. 5.35 in terms of stress and strain in a second orthogonal coordinate system as

$$R_{ki}R_{lj}\bar{\sigma}_{kl} = c_{ijkl}R_{mk}R_{nl}\bar{\varepsilon}_{mn}. \tag{5.36}$$

If we multiply both sides of this equation by $R_{pj}R_{oi}$, and sum repeated indices, we obtain

$$R_{pj}R_{oi}R_{ki}R_{lj}\bar{\sigma}_{kl} = R_{oi}R_{pj}R_{mk}R_{nl}c_{ijkl}\bar{\varepsilon}_{mn}, \tag{5.37}$$

or, because $\mathbf{R}$ is an orthogonal matrix,

$$\delta_{ok}\delta_{pl}\bar{\sigma}_{kl} = \bar{\sigma}_{op} = R_{oi}R_{pj}R_{mk}R_{nl}c_{ijkl}\bar{\varepsilon}_{mn}. \tag{5.38}$$

Since, in the second coordinate system,

$$\bar{\sigma}_{op} = \bar{c}_{opmn}\bar{\varepsilon}_{mn}, \tag{5.39}$$

we conclude that

$$\bar{c}_{opmn} = R_{oi}R_{pj}R_{mk}R_{nl}c_{ijkl}, \tag{5.40}$$

which proves that c_{ijkl} is a tensor of rank 4.

3. Stiffness matrix in finite element analysis

In the finite element method for linear analysis of structures, the forces $\mathbf{F}$ acting on an object in static equilibrium are a linear combination of the displacements $\mathbf{u}$ (or *vice versa*):

$$\mathbf{K}\mathbf{u} = \mathbf{F}, \tag{5.41}$$

where $\mathbf{K}$ is referred to as the stiffness matrix (with dimensions of force/displacement). In this equation, $\mathbf{u}$ and $\mathbf{F}$ contain several subvectors, since $\mathbf{u}$ and $\mathbf{F}$ are the displacement and force vectors for all grid points, i.e.,

$$\mathbf{u} = \begin{Bmatrix} \mathbf{u}_a \\ \mathbf{u}_b \\ \mathbf{u}_c \\ \vdots \end{Bmatrix}, \quad \mathbf{F} = \begin{Bmatrix} \mathbf{F}_a \\ \mathbf{F}_b \\ \mathbf{F}_c \\ \vdots \end{Bmatrix} \tag{5.42}$$

for grid points $a, b, c, \ldots$, where

$$\bar{\mathbf{u}}_a = \mathbf{R}_a\mathbf{u}_a, \quad \bar{\mathbf{u}}_b = \mathbf{R}_b\mathbf{u}_b, \quad \cdots . \tag{5.43}$$

Thus,

$$\bar{\mathbf{u}} = \begin{Bmatrix} \bar{\mathbf{u}}_a \\ \bar{\mathbf{u}}_b \\ \bar{\mathbf{u}}_c \\ \vdots \end{Bmatrix} = \begin{bmatrix} \mathbf{R}_a & \mathbf{0} & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{R}_b & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{0} & \mathbf{R}_c & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{Bmatrix} \mathbf{u}_a \\ \mathbf{u}_b \\ \mathbf{u}_c \\ \vdots \end{Bmatrix} = \mathbf{\Gamma}\mathbf{u}, \tag{5.44}$$

where $\mathbf{\Gamma}$ is an orthogonal block-diagonal matrix consisting of rotation matrices $\mathbf{R}$:

$$\mathbf{\Gamma} = \begin{bmatrix} \mathbf{R}_a & \mathbf{0} & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{R}_b & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{0} & \mathbf{R}_c & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}, \tag{5.45}$$

and

$$\mathbf{\Gamma}^T\mathbf{\Gamma} = \mathbf{I}. \tag{5.46}$$

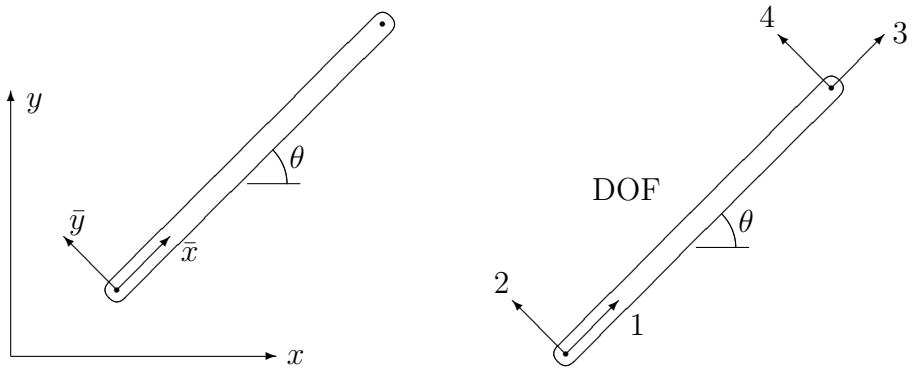


Figure 50: Element Coordinate System for Pin-Jointed Rod.

Similarly,

$$\bar{\mathbf{F}} = \mathbf{\Gamma F}. \tag{5.47}$$

Thus, if

$$\bar{\mathbf{K}}\bar{\mathbf{u}} = \bar{\mathbf{F}}, \tag{5.48}$$

$$\bar{\mathbf{K}}\mathbf{\Gamma u} = \mathbf{\Gamma F} \tag{5.49}$$

or

$$\left(\mathbf{\Gamma}^T\bar{\mathbf{K}}\mathbf{\Gamma}\right)\mathbf{u} = \mathbf{F}. \tag{5.50}$$

That is, the stiffness matrix transforms like other tensors of rank 2:

$$\mathbf{K} = \mathbf{\Gamma}^T\bar{\mathbf{K}}\mathbf{\Gamma}. \tag{5.51}$$

We illustrate the transformation of a finite element stiffness matrix by transforming the stiffness matrix for the pin-jointed rod element from a local element coordinate system to a global Cartesian system. Consider the rod shown in Fig. 50. For this element, the 4×4 2-D stiffness matrix in the element coordinate system is

$$\bar{\mathbf{K}} = \frac{AE}{L}\begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{5.52}$$

where A is the cross-sectional area of the rod, E is Young's modulus of the rod material, and L is the rod length. In the global coordinate system,

$$\mathbf{K} = \mathbf{\Gamma}^T\bar{\mathbf{K}}\mathbf{\Gamma}, \tag{5.53}$$

where the 4×4 transformation matrix is

$$\mathbf{\Gamma} = \begin{bmatrix} \mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{bmatrix}, \tag{5.54}$$

and the rotation matrix is

$$\mathbf{R} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}. \tag{5.55}$$

Thus,

$$\mathbf{K} = \frac{AE}{L}\begin{bmatrix} c & -s & 0 & 0 \\ s & c & 0 & 0 \\ 0 & 0 & c & -s \\ 0 & 0 & s & c \end{bmatrix}\begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} c & s & 0 & 0 \\ -s & c & 0 & 0 \\ 0 & 0 & c & s \\ 0 & 0 & -s & c \end{bmatrix}$$

$$= \frac{AE}{L}\begin{bmatrix} c^2 & cs & -c^2 & -cs \\ cs & s^2 & -cs & -s^2 \\ -c^2 & -cs & c^2 & cs \\ -cs & -s^2 & cs & s^2 \end{bmatrix}, \tag{5.56}$$

where $c = \cos\theta$ and $s = \sin\theta$. This result agrees with that found earlier in Eq. 4.27.

5.3 Isotropic Tensors

An *isotropic tensor* is a tensor which is independent of coordinate system (i.e., invariant under an orthogonal coordinate transformation). The Kronecker delta δ_{ij} is a second rank tensor and isotropic, since $\bar{\delta}_{ij} = \delta_{ij}$, and

$$\bar{\mathbf{I}} = \mathbf{R}\mathbf{I}\mathbf{R}^T = \mathbf{R}\mathbf{R}^T = \mathbf{I}. \tag{5.57}$$

That is, the identity matrix $\mathbf{I}$ is invariant under an orthogonal coordinate transformation.

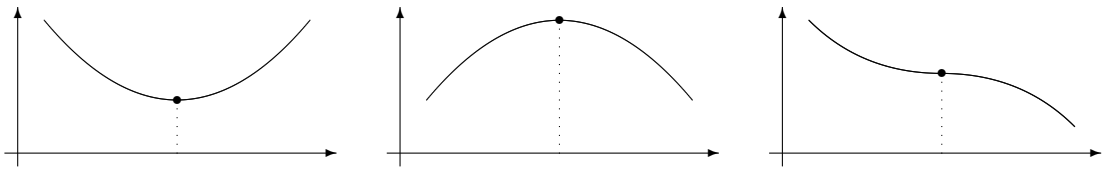
It can be shown in tensor analysis that δ_{ij} is the only isotropic tensor of rank 2 and, moreover, δ_{ij} is the characteristic tensor for all isotropic tensors:

Rank	Isotropic Tensors
1	none
2	$c\delta_{ij}$
3	none
4	$a\delta_{ij}\delta_{kl} + b\delta_{ik}\delta_{jl} + c\delta_{il}\delta_{jk}$
odd	none

That is, all isotropic tensors of rank 4 must be of the form shown above, which has three constants. For example, in generalized Hooke's law (Eq. 5.35), the material property tensor c_{ijkl} has $3^4 = 81$ constants (assuming no symmetry). For an isotropic material, c_{ijkl} must be an isotropic tensor of rank 4, thus implying at most three material constants (on the basis of tensor analysis alone). The actual number of independent material constants for an isotropic material turns out to be two rather than three, a result which depends on the existence of a strain energy function, which implies the additional symmetry $c_{ijkl} = c_{klij}$.

6 Calculus of Variations

Recall from calculus that a function of one variable $y = f(x)$ attains a stationary value (minimum, maximum, or neutral) at the point $x = x_0$ if the derivative $f'(x) = 0$ at $x = x_0$.



(a) Minimum (b) Maximum (c) Neutral

Figure 51: Minimum, Maximum, and Neutral Stationary Values.

Alternatively, the *first variation* of f (which is similar to a differential) must vanish for arbitrary changes δx in x:

$$\delta f = \left(\frac{df}{dx}\right)\delta x = 0. \tag{6.1}$$

The second derivative determines what kind of stationary value one has:

$$\text{minimum: } \frac{d^2f}{dx^2} > 0 \text{ at } x = x_0 \tag{6.2}$$

$$\text{maximum: } \frac{d^2f}{dx^2} < 0 \text{ at } x = x_0 \tag{6.3}$$

$$\text{neutral: } \frac{d^2f}{dx^2} = 0 \text{ at } x = x_0, \tag{6.4}$$

as shown in Fig. 51.

For a function of two variables $z = f(x, y)$, z is stationary at (x_0, y_0) if

$$\frac{\partial f}{\partial x} = 0 \text{ and } \frac{\partial f}{\partial y} = 0 \text{ at } (x_0, y_0).$$

Alternatively, for a stationary value,

$$\delta f = \left(\frac{\partial f}{\partial x}\right)\delta x + \left(\frac{\partial f}{\partial y}\right)\delta y = 0 \text{ at } (x_0, y_0).$$

A function with n independent variables $f(x_1, x_2, \ldots, x_n)$ is stationary at a point P if

$$\delta f = \sum_{i=1}^{n} \frac{\partial f}{\partial x_i}\delta x_i = 0 \text{ at } P$$

for arbitrary variations δx_i. Hence,

$$\frac{\partial f}{\partial x_i} = 0 \text{ at } P \quad (i = 1, 2, \ldots, n).$$

Variational calculus is concerned with finding stationary values, not of functions, but of functionals. In general, a *functional* is a function of a function. More precisely, a functional is an integral which has a specific numerical value for each function that is substituted into it.

Consider the functional

$$I(\phi) = \int_a^b F(x, \phi, \phi_x)\, dx, \tag{6.5}$$

where x is the independent variable, $\phi(x)$ is the dependent variable, and

$$\phi_x = \frac{d\phi}{dx}. \tag{6.6}$$

The variation in I due to an arbitrary small change in ϕ is

$$\delta I = \int_a^b \delta F\, dx = \int_a^b \left(\frac{\partial F}{\partial \phi} \delta\phi + \frac{\partial F}{\partial \phi_x} \delta\phi_x \right) dx, \tag{6.7}$$

where, with an interchange of the order of d and δ,

$$\delta\phi_x = \delta \left(\frac{d\phi}{dx} \right) = \frac{d}{dx}(\delta\phi). \tag{6.8}$$

With this equation, the second term in Eq. 6.7 can be integrated by parts to obtain

$$\int_a^b \frac{\partial F}{\partial \phi_x} \delta\phi_x\, dx = \int_a^b \frac{\partial F}{\partial \phi_x} \frac{d}{dx}(\delta\phi)\, dx = \left. \frac{\partial F}{\partial \phi_x} \delta\phi \right|_a^b - \int_a^b \delta\phi\, \frac{d}{dx} \left(\frac{\partial F}{\partial \phi_x} \right) dx. \tag{6.9}$$

Thus,

$$\delta I = \int_a^b \left[\frac{\partial F}{\partial \phi} - \frac{d}{dx} \left(\frac{\partial F}{\partial \phi_x} \right) \right] \delta\phi\, dx + \left. \left(\frac{\partial F}{\partial \phi_x} \delta\phi \right) \right|_a^b. \tag{6.10}$$

Since $\delta\phi$ is arbitrary (within certain limits of admissibility), $\delta I = 0$ implies that both terms in Eq. 6.10 must vanish. Moreover, for arbitrary $\delta\phi$, a zero integral in Eq. 6.10 implies a zero integrand:

$$\frac{d}{dx} \left(\frac{\partial F}{\partial \phi_x} \right) - \frac{\partial F}{\partial \phi} = 0, \tag{6.11}$$

which is referred to as the *Euler-Lagrange* equation.

The second term in Eq. 6.10 must also vanish for arbitrary $\delta\phi$:

$$\left. \left(\frac{\partial F}{\partial \phi_x} \delta\phi \right) \right|_a^b = 0. \tag{6.12}$$

If ϕ is specified at the boundaries $x = a$ and $x = b$,

$$\delta\phi(a) = \delta\phi(b) = 0, \tag{6.13}$$

and Eq. 6.12 is automatically satisfied. This type of boundary condition is called a *geometric* or *essential* boundary condition. If ϕ is not specified at the boundaries $x = a$ and $x = b$, Eq. 6.12 implies

$$\frac{\partial F(a)}{\partial \phi_x} = \frac{\partial F(b)}{\partial \phi_x} = 0. \tag{6.14}$$

This type of boundary condition is called a *natural* boundary condition.

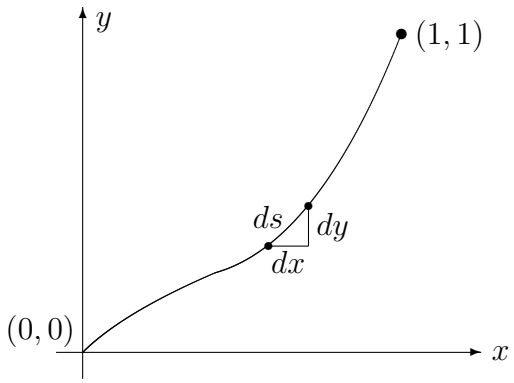


Figure 52: Curve of Minimum Length Between Two Points.

6.1 Example 1: The Shortest Distance Between Two Points

We wish to find the equation of the curve $y(x)$ along which the distance from the origin $(0, 0)$ to $(1, 1)$ in the xy-plane is least. Consider the curve in Fig. 52. The differential arc length along the curve is given by

$$ds = \sqrt{dx^2 + dy^2} = \sqrt{1 + (y')^2}\, dx. \tag{6.15}$$

We seek the curve $y(x)$ which minimizes the total arc length

$$I(y) = \int_0^1 \sqrt{1 + (y')^2}\, dx, \tag{6.16}$$

where $y(0) = 0$ and $y(1) = 1$. For this problem,

$$F(x, y, y') = \left[1 + (y')^2\right]^{1/2}, \tag{6.17}$$

which depends only on y' explicitly. Thus, the Euler-Lagrange equation, Eq. 6.11, is

$$\frac{d}{dx}\left(\frac{\partial F}{\partial y'}\right) = 0, \tag{6.18}$$

where

$$\frac{\partial F}{\partial y'} = \frac{1}{2}\left[1 + (y')^2\right]^{-1/2} 2y' = \frac{y'}{\left[1 + (y')^2\right]^{1/2}}. \tag{6.19}$$

Hence,

$$\frac{y'}{\left[1 + (y')^2\right]^{1/2}} = c, \tag{6.20}$$

where c is a constant of integration. If we solve this equation for y', we obtain

$$y' = \sqrt{\frac{c^2}{1 - c^2}} = a, \tag{6.21}$$

where a is another constant. Thus,

$$y(x) = ax + b, \tag{6.22}$$

and, with the boundary conditions,

$$y(x) = x, \tag{6.23}$$

which is the straight line joining $(0, 0)$ and $(1, 1)$, as expected.

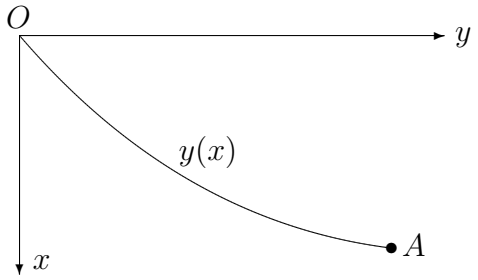


Figure 53: The Brachistochrone Problem.

6.2 Example 2: The Brachistochrone

We wish to determine the path $y(x)$ from the origin O to the point A along which a bead will slide under the force of gravity and without friction in the shortest time (Fig. 53). Assume that the bead starts from rest. The velocity v along the path $y(x)$ is

$$v = \frac{ds}{dt} = \frac{\sqrt{dx^2 + dy^2}}{dt} = \frac{\sqrt{1 + (y')^2}\, dx}{dt}, \tag{6.24}$$

where s denotes distance along the path. Thus,

$$dt = \frac{\sqrt{1 + (y')^2}\, dx}{v}. \tag{6.25}$$

As the bead slides down the path, potential energy is converted to kinetic energy. At any location x, the kinetic energy equals the reduction in potential energy:

$$\frac{1}{2}mv^2 = mgx \tag{6.26}$$

or

$$v = \sqrt{2gx}. \tag{6.27}$$

Thus, from Eq. 6.25,

$$dt = \sqrt{\frac{1 + (y')^2}{2gx}}\, dx. \tag{6.28}$$

The total time for the bead to fall from O to A is then

$$t = \int_0^{x_A} \sqrt{\frac{1 + (y')^2}{2gx}}\, dx, \tag{6.29}$$

which is the integral to be minimized. From Eq. 6.11, the Euler-Lagrange equation is

$$\frac{d}{dx}\left(\frac{\partial F}{\partial y'}\right) - \frac{\partial F}{\partial y} = 0, \tag{6.30}$$

where

$$F(x, y, y') = \sqrt{\frac{1 + (y')^2}{2gx}}. \tag{6.31}$$

Thus, the Euler-Lagrange equation becomes

$$\frac{d}{dx}\left[\frac{1}{2}\left(\frac{1+(y')^2}{2gx}\right)^{-1/2}\frac{2y'}{2gx}\right]=0 \tag{6.32}$$

or

$$\frac{d}{dx}\left(\frac{y'}{\{x\,[1+(y')^2]\}^{1/2}}\right)=0. \tag{6.33}$$

To solve this equation, we integrate Eq. 6.33 to obtain

$$\frac{y'}{\{x\,[1+(y')^2]\}^{1/2}}=c, \tag{6.34}$$

where c is a constant of integration. We then square both sides of this equation, and solve for y':

$$(y')^2=c^2x\left[1+(y')^2\right] \tag{6.35}$$

or

$$y'=\frac{dy}{dx}=\sqrt{\frac{c^2x}{1-c^2x}}. \tag{6.36}$$

This equation can be integrated using the change of variable

$$x=\frac{1}{c^2}\sin^2(\theta/2),\quad dx=\frac{1}{c^2}\sin(\theta/2)\cos(\theta/2)\,d\theta. \tag{6.37}$$

The integral implied by Eq. 6.36 then transforms to

$$y=\int\frac{\sin(\theta/2)}{\cos(\theta/2)}\cdot\frac{1}{c^2}\sin(\theta/2)\cos(\theta/2)\,d\theta=\frac{1}{c^2}\int\sin^2(\theta/2)\,d\theta \tag{6.38}$$

$$=\frac{1}{2c^2}\int(1-\cos\theta)\,d\theta=\frac{1}{2c^2}(\theta-\sin\theta)+y_0, \tag{6.39}$$

where y_0 is a constant of integration. Since the curve must pass through the origin, we must have $y_0=0$. Also, from Eq. 6.37,

$$x=\frac{1}{2c^2}(1-\cos\theta). \tag{6.40}$$

If we then replace the constant $a=1/(2c^2)$, we obtain the final result in parametric form

$$\begin{cases} x=a(1-\cos\theta)\\ y=a(\theta-\sin\theta), \end{cases} \tag{6.41}$$

which is a cycloid generated by the motion of a fixed point on the circumference of a circle of radius a which rolls on the positive side of the line $x=0$.

To solve Eq. 6.41 for a specific cycloid (defined by the two end points), one can first eliminate the circle radius a from Eq. 6.41 to solve (iteratively) for θ_A (the value of θ at Point A), after which either of the two equations in Eq. 6.41 can be used to determine the constant a. The resulting cycloids for several end points are shown in Fig. 54.

This brachistochrone solution is valid for any end point which the bead can reach. Thus, the end point must not be higher than the starting point. The end point may be at the same elevation since, without friction, there is no loss of energy during the motion.

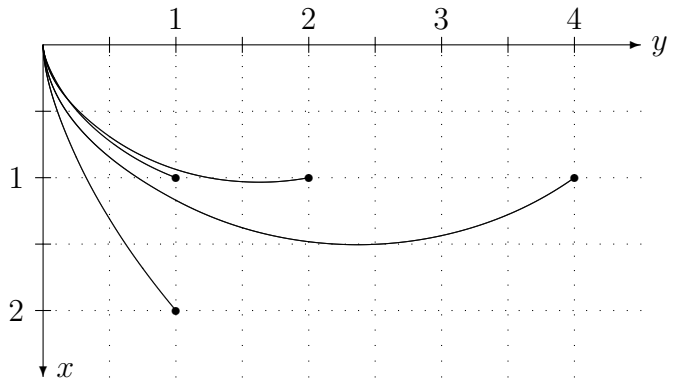


Figure 54: Several Brachistochrone Solutions.

6.3 Constraint Conditions

Suppose we want to extremize the functional

$$I(\phi) = \int_a^b F(x, \phi, \phi_x)\, dx \tag{6.42}$$

subject to the additional constraint condition

$$\int_a^b G(x, \phi, \phi_x)\, dx = J, \tag{6.43}$$

where J is a specified constant. We recall that the Euler-Lagrange equation was found by requiring that $\delta I = 0$. However, since J is a constant, $\delta J = 0$. Thus,

$$\delta(I + \lambda J) = 0, \tag{6.44}$$

where λ is an unknown constant referred to as a *Lagrange multiplier*. Thus, to enforce the constraint in Eq. 6.43, we can replace F in the Euler-Lagrange equation with

$$H = F + \lambda G. \tag{6.45}$$

6.4 Example 3: A Constrained Minimization Problem

We wish to find the function $y(x)$ which minimizes the integral

$$I(y) = \int_0^1 (y')^2\, dx \tag{6.46}$$

subject to the end conditions $y(0) = y(1) = 0$ and the constraint

$$\int_0^1 y\, dx = 1. \tag{6.47}$$

That is, the area under the curve $y(x)$ is unity (Fig. 55). For this problem,

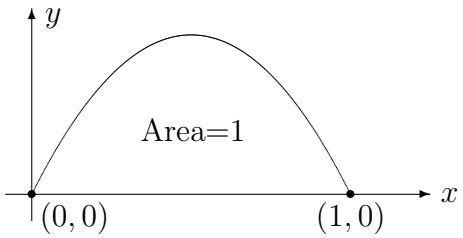


Figure 55: A Constrained Minimization Problem.

$$H(x, y, y') = (y')^2 + \lambda y. \tag{6.48}$$

The Euler-Lagrange equation,

$$\frac{d}{dx}\left(\frac{\partial H}{\partial y'}\right) - \frac{\partial H}{\partial y} = 0, \tag{6.49}$$

yields

$$\frac{d}{dx}(2y') - \lambda = 0 \tag{6.50}$$

or

$$y'' = \lambda/2. \tag{6.51}$$

After integrating this equation, we obtain

$$y = \frac{\lambda}{4}x^2 + Ax + B. \tag{6.52}$$

With the boundary conditions $y(0) = y(1) = 0$, we obtain $B = 0$ and $A = -\lambda/4$. Thus,

$$y = -\lambda x(1 - x)/4. \tag{6.53}$$

The area constraint, Eq. 6.47, is used to evaluate the Lagrange multiplier λ:

$$1 = \int_0^1 y\, dx = -\int_0^1 \frac{\lambda}{4}\left(x - x^2\right) dx = -\frac{\lambda}{4}\left(\frac{x^2}{2} - \frac{x^3}{3}\right)\Bigg|_0^1 = -\frac{\lambda}{24}. \tag{6.54}$$

Thus, with $\lambda = -24$, we obtain the parabola

$$y(x) = 6x(1 - x). \tag{6.55}$$

6.5 Functions of Several Independent Variables

Consider

$$I(\phi) = \int_V F(x, y, z, \phi, \phi_x, \phi_y, \phi_z)\, dV, \tag{6.56}$$

a functional with three independent variables (x, y, z). Note that, in 3-D, the integration is a volume integral.

The variation in I due to an arbitrary small change in ϕ is

$$\delta I = \int_V \left(\frac{\partial F}{\partial \phi}\delta\phi + \frac{\partial F}{\partial \phi_x}\delta\phi_x + \frac{\partial F}{\partial \phi_y}\delta\phi_y + \frac{\partial F}{\partial \phi_z}\delta\phi_z\right) dV, \tag{6.57}$$

where, with an interchange of the order of ∂ and δ,

$$\delta I = \int_V \left[\frac{\partial F}{\partial \phi}\delta\phi + \frac{\partial F}{\partial \phi_x}\frac{\partial}{\partial x}(\delta\phi) + \frac{\partial F}{\partial \phi_y}\frac{\partial}{\partial y}(\delta\phi) + \frac{\partial F}{\partial \phi_z}\frac{\partial}{\partial z}(\delta\phi) \right] dV. \tag{6.58}$$

To perform a three-dimensional integration by parts, we note that the second term in Eq. 6.58 can be expanded to yield

$$\frac{\partial F}{\partial \phi_x}\frac{\partial}{\partial x}(\delta\phi) = \frac{\partial}{\partial x}\left(\frac{\partial F}{\partial \phi_x}\delta\phi\right) - \frac{\partial}{\partial x}\left(\frac{\partial F}{\partial \phi_x}\right)\delta\phi. \tag{6.59}$$

If we then expand the third and fourth terms similarly, Eq. 6.58 becomes, after regrouping,

$$\begin{aligned}\delta I = \int_V \Bigg[&\frac{\partial F}{\partial \phi}\delta\phi - \frac{\partial}{\partial x}\left(\frac{\partial F}{\partial \phi_x}\right)\delta\phi - \frac{\partial}{\partial y}\left(\frac{\partial F}{\partial \phi_y}\right)\delta\phi - \frac{\partial}{\partial z}\left(\frac{\partial F}{\partial \phi_z}\right)\delta\phi \\ &+ \frac{\partial}{\partial x}\left(\frac{\partial F}{\partial \phi_x}\delta\phi\right) + \frac{\partial}{\partial y}\left(\frac{\partial F}{\partial \phi_y}\delta\phi\right) + \frac{\partial}{\partial z}\left(\frac{\partial F}{\partial \phi_z}\delta\phi\right) \Bigg] dV. \end{aligned} \tag{6.60}$$

The last three terms in this integral can then be converted to a surface integral using the divergence theorem, which states that, for a vector field $\mathbf{f}$,

$$\int_V \nabla \cdot \mathbf{f}\, dV = \oint_S \mathbf{f} \cdot \mathbf{n}\, dS, \tag{6.61}$$

where S is the closed surface which encloses the volume V. Eq. 6.60 then becomes

$$\begin{aligned}\delta I = &\int_V \left[\frac{\partial F}{\partial \phi} - \frac{\partial}{\partial x}\left(\frac{\partial F}{\partial \phi_x}\right) - \frac{\partial}{\partial y}\left(\frac{\partial F}{\partial \phi_y}\right) - \frac{\partial}{\partial z}\left(\frac{\partial F}{\partial \phi_z}\right) \right] \delta\phi\, dV \\ &+ \oint_S \left(\frac{\partial F}{\partial \phi_x}n_x + \frac{\partial F}{\partial \phi_y}n_y + \frac{\partial F}{\partial \phi_z}n_z \right) \delta\phi\, dS, \end{aligned} \tag{6.62}$$

where $\mathbf{n} = (n_x, n_y, n_z)$ is the unit outward normal on the surface.

Since $\delta I = 0$, both integrals in this equation must vanish for arbitrary $\delta\phi$. It therefore follows that the integrand in the volume integral must also vanish to yield the Euler-Lagrange equation for three independent variables:

$$\frac{\partial}{\partial x}\left(\frac{\partial F}{\partial \phi_x}\right) + \frac{\partial}{\partial y}\left(\frac{\partial F}{\partial \phi_y}\right) + \frac{\partial}{\partial z}\left(\frac{\partial F}{\partial \phi_z}\right) - \frac{\partial F}{\partial \phi} = 0. \tag{6.63}$$

If ϕ is specified on the boundary S, $\delta\phi = 0$ on S, and the boundary integral in Eq. 6.62 automatically vanishes. If ϕ is not specified on S, the integrand in the boundary integral must vanish:

$$\frac{\partial F}{\partial \phi_x}n_x + \frac{\partial F}{\partial \phi_y}n_y + \frac{\partial F}{\partial \phi_z}n_z = 0 \quad \text{on } S. \tag{6.64}$$

This is the natural boundary condition when ϕ is not specified on the boundary.

6.6 Example 4: Poisson's Equation

Consider the functional

$$I(\phi) = \int_V \left[\frac{1}{2}(\phi_x^2 + \phi_y^2 + \phi_z^2) - Q\phi \right] dV, \tag{6.65}$$

in which case

$$F(x, y, z, \phi, \phi_x, \phi_y, \phi_z) = \frac{1}{2}(\phi_x^2 + \phi_y^2 + \phi_z^2) - Q\phi. \tag{6.66}$$

The Euler-Lagrange equation, Eq. 6.63, implies

$$\frac{\partial}{\partial x}(\phi_x) + \frac{\partial}{\partial y}(\phi_y) + \frac{\partial}{\partial z}(\phi_z) - (-Q) = 0. \tag{6.67}$$

That is,

$$\phi_{xx} + \phi_{yy} + \phi_{zz} = -Q \tag{6.68}$$

or

$$\nabla^2 \phi = -Q, \tag{6.69}$$

which is Poisson's equation. Thus, we have shown that solving Poisson's equation is equivalent to minimizing the functional I in Eq. 6.65. In general, a key conclusion of this discussion of variational calculus is that solving a differential equation is equivalent to minimizing some functional involving an integral.

6.7 Functions of Several Dependent Variables

Consider the functional

$$I(\phi_1, \phi_2, \phi_3) = \int_a^b F(x, \phi_1, \phi_2, \phi_3, \phi_1', \phi_2', \phi_3')\, dx, \tag{6.70}$$

where x is the independent variable, and $\phi_1(x)$, $\phi_2(x)$, and $\phi_3(x)$ are the dependent variables. It can be shown that the generalization of the Euler-Lagrange equation, Eq. 6.11, for this case is

$$\frac{d}{dx}\left(\frac{\partial F}{\partial \phi_1'}\right) - \frac{\partial F}{\partial \phi_1} = 0 \tag{6.71}$$

$$\frac{d}{dx}\left(\frac{\partial F}{\partial \phi_2'}\right) - \frac{\partial F}{\partial \phi_2} = 0 \tag{6.72}$$

$$\frac{d}{dx}\left(\frac{\partial F}{\partial \phi_3'}\right) - \frac{\partial F}{\partial \phi_3} = 0. \tag{6.73}$$

7 Variational Approach to the Finite Element Method

In modern engineering analysis, one of the most important applications of the energy theorems, such as the minimum potential energy theorem in elasticity, is the finite element method, a numerical procedure for solving partial differential equations. For linear equations, finite element solutions are often based on variational methods.

7.1 Index Notation and Summation Convention

Let $\mathbf{a}$ be the vector

$$\mathbf{a} = a_1\mathbf{e}_1 + a_2\mathbf{e}_2 + a_3\mathbf{e}_3, \tag{7.1}$$

where $\mathbf{e}_i$ is the unit vector in the ith Cartesian coordinate direction, and a_i is the ith component of $\mathbf{a}$ (the projection of $\mathbf{a}$ on $\mathbf{e}_i$). Consider a second vector

$$\mathbf{b} = b_1\mathbf{e}_1 + b_2\mathbf{e}_2 + b_3\mathbf{e}_3. \tag{7.2}$$

Then, the dot product (scalar product) is

$$\mathbf{a}\cdot\mathbf{b} = a_1b_1 + a_2b_2 + a_3b_3 = \sum_{i=1}^{3} a_ib_i. \tag{7.3}$$

Using the *summation convection*, we can omit the summation symbol and write

$$\mathbf{a}\cdot\mathbf{b} = a_ib_i, \tag{7.4}$$

where, if a subscript appears exactly twice, a summation is implied over the range. The range is 1,2,3 in 3-D and 1,2 in 2-D.

Let $f(x_1, x_2, x_3)$ be a scalar function of the Cartesian coordinates x_1, x_2, x_3. We define the shorthand comma notation for derivatives:

$$\frac{\partial f}{\partial x_i} = f_{,i}. \tag{7.5}$$

That is, the subscript "$,i$" denotes the partial derivative with respect to the ith Cartesian coordinate direction.

Using the comma notation and the summation convention, a variety of familiar quantities can be written in compact form. For example, the gradient can be written

$$\nabla f = \frac{\partial f}{\partial x_1}\mathbf{e}_1 + \frac{\partial f}{\partial x_2}\mathbf{e}_2 + \frac{\partial f}{\partial x_3}\mathbf{e}_3 = f_{,i}\mathbf{e}_i. \tag{7.6}$$

For a vector-valued function $\mathbf{F}(x_1, x_2, x_3)$, the divergence is written

$$\nabla\cdot\mathbf{F} = \frac{\partial F_1}{\partial x_1} + \frac{\partial F_2}{\partial x_2} + \frac{\partial F_3}{\partial x_3} = F_{i,i}, \tag{7.7}$$

and the Laplacian of the scalar function $f(x_1, x_2, x_3)$ is

$$\nabla^2 f = \nabla\cdot\nabla f = \frac{\partial^2 f}{\partial x_1^2} + \frac{\partial^2 f}{\partial x_2^2} + \frac{\partial^2 f}{\partial x_3^2} = f_{,ii}. \tag{7.8}$$

The divergence theorem states that, for any vector field $\mathbf{F}(x_1, x_2, x_3)$ defined in a volume V bounded by a closed surface S,

$$\int_V \nabla\cdot\mathbf{F}\, dV = \oint_S \mathbf{F}\cdot\mathbf{n}\, dS \tag{7.9}$$

or, in index notation,

$$\int_V F_{i,i}\, dV = \oint_S F_i n_i\, dS. \tag{7.10}$$

The normal derivative can be written in index notation as

$$\frac{\partial \phi}{\partial n} = \nabla\phi \cdot \mathbf{n} = \phi_{,i} n_i. \tag{7.11}$$

The dot product of two gradients can be written

$$\nabla\phi \cdot \nabla\phi = (\phi_{,i}\mathbf{e}_i) \cdot (\phi_{,j}\mathbf{e}_j) = \phi_{,i}\phi_{,j}\, \mathbf{e}_i \cdot \mathbf{e}_j = \phi_{,i}\phi_{,j}\, \delta_{ij} = \phi_{,i}\phi_{,i}, \tag{7.12}$$

where δ_{ij} is the *Kronecker delta* defined as

$$\delta_{ij} = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases} \tag{7.13}$$

7.2 Deriving Variational Principles

For each partial differential equation of interest, one needs a functional which a solution makes stationary. Given the functional, it is generally easy to see what partial differential equation corresponds to it. The harder problem is to start with the equation and derive the variational principle (i.e., derive the functional which is to be minimized). To simplify this discussion, we will consider only scalar, rather than vector, problems. A scalar problem has one dependent variable and one partial differential equation, whereas a vector problem has several dependent variables coupled to each other through a system of partial differential equations.

Consider Poisson's equation subject to both Dirichlet and Neumann boundary conditions,

$$\begin{cases} \nabla^2\phi + f = 0 \ \text{ in } V, \\ \phi = \phi_0 \ \text{ on } S_1, \\ \dfrac{\partial \phi}{\partial n} + g = 0 \ \text{ on } S_2. \end{cases} \tag{7.14}$$

This problem arises, for example, in (1) steady-state heat conduction, where the temperature and heat flux are specified on different parts of the boundary, (2) potential fluid flow, where the velocity potential and velocity are specified on different parts of the boundary, and (3) torsion in elasticity. Recall that, in index notation, $\nabla^2\phi = \phi_{,ii}$.

We wish to find a functional, $I(\phi)$ say, whose first variation δI vanishes for ϕ satisfying the above boundary value problem. From Eq. 7.14a,

$$0 = \int_V (\phi_{,ii} + f)\, \delta\phi\, dV \tag{7.15}$$

$$= \int_V [(\phi_{,i}\, \delta\phi)_{,i} - \phi_{,i}\, \delta\phi_{,i}]\, dV + \int_V \delta(f\phi)\, dV \tag{7.16}$$

$$= \oint_S \phi_{,i} n_i\, \delta\phi\, dS - \int_V \frac{1}{2}\, \delta(\phi_{,i}\phi_{,i})\, dV + \delta \int_V f\phi\, dV \tag{7.17}$$

$$= \oint_S (\nabla\phi \cdot \mathbf{n})\, \delta\phi\, dS - \delta \int_V \frac{1}{2}\phi_{,i}\phi_{,i}\, dV + \delta \int_V f\phi\, dV, \tag{7.18}$$

where $\nabla\phi \cdot \mathbf{n} = \partial\phi/\partial n = -g$ on S_2, and, since ϕ is specified on S_1, $\delta\phi = 0$ on S_1. Then,

$$0 = -\int_{S_2} g\,\delta\phi\,dS - \delta\int_V \frac{1}{2}\phi_{,i}\phi_{,i}\,dV + \delta\int_V f\phi\,dV \tag{7.19}$$

$$= -\delta\int_{S_2} g\phi\,dS - \delta\int_V \frac{1}{2}\phi_{,i}\phi_{,i}\,dV + \delta\int_V f\phi\,dV \tag{7.20}$$

$$= -\delta\left[\int_V \left(\frac{1}{2}\phi_{,i}\phi_{,i} - f\phi\right)dV + \int_{S_2} g\phi\,dS\right] = -\delta I(\phi). \tag{7.21}$$

Thus, the functional for this boundary value problem is

$$I(\phi) = \int_V \left(\frac{1}{2}\phi_{,i}\phi_{,i} - f\phi\right)dV + \int_{S_2} g\phi\,dS \tag{7.22}$$

or, in vector notation,

$$I(\phi) = \int_V \left(\frac{1}{2}\nabla\phi\cdot\nabla\phi - f\phi\right)dV + \int_{S_2} g\phi\,dS \tag{7.23}$$

or, in expanded form,

$$I(\phi) = \int_V \left\{\frac{1}{2}\left[\left(\frac{\partial\phi}{\partial x}\right)^2 + \left(\frac{\partial\phi}{\partial y}\right)^2 + \left(\frac{\partial\phi}{\partial z}\right)^2\right] - f\phi\right\}dV + \int_{S_2} g\phi\,dS. \tag{7.24}$$

If we were given the functional, Eq. 7.22, we could determine which partial differential equation corresponds to it by computing and setting to zero the first variation of the functional. From Eq. 7.22,

$$\delta I = \int_V \left(\frac{1}{2}\,\delta\phi_{,i}\phi_{,i} + \frac{1}{2}\phi_{,i}\,\delta\phi_{,i} - f\,\delta\phi\right)dV + \int_{S_2} g\,\delta\phi\,dS \tag{7.25}$$

$$= \int_V (\phi_{,i}\,\delta\phi_{,i} - f\delta\phi)\,dV + \int_{S_2} g\,\delta\phi\,dS \tag{7.26}$$

$$= \int_V [(\phi_{,i}\delta\phi)_{,i} - \phi_{,ii}\,\delta\phi]\,dV - \int_V f\,\delta\phi\,dV + \int_{S_2} g\,\delta\phi\,dS \tag{7.27}$$

$$= \oint_S \phi_{,i}n_i\,\delta\phi\,dS - \int_V (\phi_{,ii} + f)\,\delta\phi\,dV + \int_{S_2} g\,\delta\phi\,dS, \tag{7.28}$$

where $\delta\phi = 0$ on S_1, and $\phi_{,i}n_i = \nabla\phi\cdot\mathbf{n} = \partial\phi/\partial n$. Hence,

$$\delta I = \int_{S_2} \left(\frac{\partial\phi}{\partial n} + g\right)\delta\phi\,dS - \int_V (\phi_{,ii} + f)\,\delta\phi\,dV. \tag{7.29}$$

Stationary I ($\delta I = 0$) for arbitrary admissible $\delta\phi$ thus yields the original partial differential equation and Neumann boundary condition, Eq. 7.14.

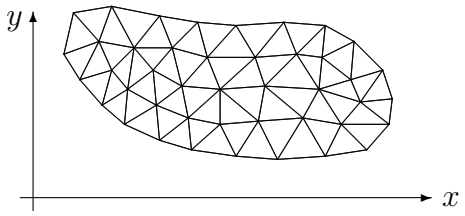


Figure 56: Two-Dimensional Finite Element Mesh.

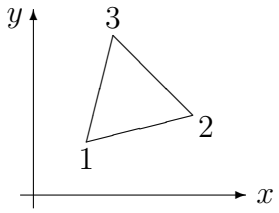


Figure 57: Triangular Finite Element.

7.3 Shape Functions

Consider a two-dimensional field problem for which we seek the function $\phi(x, y)$ satisfying some (unspecified) partial differential equation in a domain. Let us subdivide the domain into triangular finite elements, as shown in Fig. 56. A typical element, shown in Fig. 57, has three degrees of freedom (DOF): ϕ_1, ϕ_2, and ϕ_3. The three DOF are the values of the fundamental unknown ϕ at the three grid points. (The number of DOF for an element represents the number of pieces of data needed to determine uniquely the solution for the element.)

For numerical solution, we approximate $\phi(x, y)$ using a polynomial in two variables having the same number of terms as the number of DOF. Thus, we assume, for each element,

$$\phi(x, y) = a_1 + a_2 x + a_3 y, \tag{7.30}$$

where a_1, a_2, and a_3 are unknown coefficients. Eq. 7.30 is a complete linear polynomial in two variables. We can evaluate this equation at the three grid points to obtain

$$\left\{ \begin{array}{c} \phi_1 \\ \phi_2 \\ \phi_3 \end{array} \right\} = \left[\begin{array}{ccc} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{array} \right] \left\{ \begin{array}{c} a_1 \\ a_2 \\ a_3 \end{array} \right\}. \tag{7.31}$$

This matrix equation can be inverted to yield

$$\left\{ \begin{array}{c} a_1 \\ a_2 \\ a_3 \end{array} \right\} = \frac{1}{2A} \left[\begin{array}{ccc} x_2 y_3 - x_3 y_2 & x_3 y_1 - x_1 y_3 & x_1 y_2 - x_2 y_1 \\ y_2 - y_3 & y_3 - y_1 & y_1 - y_2 \\ x_3 - x_2 & x_1 - x_3 & x_2 - x_1 \end{array} \right] \left\{ \begin{array}{c} \phi_1 \\ \phi_2 \\ \phi_3 \end{array} \right\}, \tag{7.32}$$

where

$$2A = \left| \begin{array}{ccc} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{array} \right| = x_2 y_3 - x_3 y_2 - x_1 y_3 + x_3 y_1 + x_1 y_2 - x_2 y_1. \tag{7.33}$$

Note that the area of the triangle is $|A|$. The determinant $2A$ is positive or negative depending on whether the grid point ordering in the triangle is counter-clockwise or clockwise, respectively. Since Eq. 7.31 is invertible, we can interpret the element DOF as either the grid point values ϕ_i, which have physical meaning, or the nonphysical polynomial coefficients a_i.

From Eq. 7.30, the scalar unknown ϕ can then be written in the matrix form

$$\phi(x,y) = \begin{bmatrix} 1 & x & y \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \\ a_3 \end{Bmatrix} \tag{7.34}$$

$$= \frac{1}{2A} \begin{bmatrix} 1 & x & y \end{bmatrix} \begin{bmatrix} x_2y_3 - x_3y_2 & x_3y_1 - x_1y_3 & x_1y_2 - x_2y_1 \\ y_2 - y_3 & y_3 - y_1 & y_1 - y_2 \\ x_3 - x_2 & x_1 - x_3 & x_2 - x_1 \end{bmatrix} \begin{Bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{Bmatrix} \tag{7.35}$$

$$= \begin{bmatrix} N_1(x,y) & N_2(x,y) & N_3(x,y) \end{bmatrix} \begin{Bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{Bmatrix} \tag{7.36}$$

or

$$\phi(x,y) = N_1(x,y)\phi_1 + N_2(x,y)\phi_2 + N_3(x,y)\phi_3 = N_i\phi_i, \tag{7.37}$$

where

$$\begin{cases} N_1(x,y) = [(x_2y_3 - x_3y_2) + (y_2 - y_3)x + (x_3 - x_2)y]/(2A) \\ N_2(x,y) = [(x_3y_1 - x_1y_3) + (y_3 - y_1)x + (x_1 - x_3)y]/(2A) \\ N_3(x,y) = [(x_1y_2 - x_2y_1) + (y_1 - y_2)x + (x_2 - x_1)y]/(2A). \end{cases} \tag{7.38}$$

Notice that all the subscripts in this equation form a cyclic permutation of the numbers $1, 2, 3$. That is, knowing N_1, we can write down N_2 and N_3 by a cyclic permutation of the subscripts. Alternatively, if we let i, j, and k denote the three grid points of the triangle, the above equation can be written in the compact form

$$N_i(x,y) = \frac{1}{2A}[(x_jy_k - x_ky_j) + (y_j - y_k)x + (x_k - x_j)y], \tag{7.39}$$

where (i,j,k) can be selected to be any cyclic permutation of $(1,2,3)$ such as $(1,2,3)$, $(2,3,1)$, or $(3,1,2)$.

In general, the functions N_i are called *shape* functions or *interpolation* functions and are used extensively in finite element theory. Eq. 7.37 implies that N_1 can be thought of as the shape function associated with Point 1, since N_1 is the form (or "shape") that ϕ would take if $\phi_1 = 1$ and $\phi_2 = \phi_3 = 0$. In general, N_i is the shape function for Point i.

From Eq. 7.37, if $\phi_1 \neq 0$, and $\phi_2 = \phi_3 = 0$,

$$\phi(x,y) = N_1(x,y)\phi_1. \tag{7.40}$$

In particular, at Point 1,

$$\phi_1 = \phi(x_1,y_1) = N_1(x_1,y_1)\phi_1 \tag{7.41}$$

or

$$N_1(x_1,y_1) = 1. \tag{7.42}$$

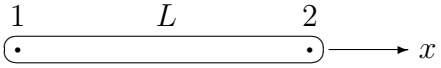


Figure 58: Axial Member (Pin-Jointed Truss Element).

Also, at Point 2,

$$0 = \phi_2 = \phi(x_2, y_2) = N_1(x_2, y_2)\phi_1 \tag{7.43}$$

or

$$N_1(x_2, y_2) = 0. \tag{7.44}$$

Similarly,

$$N_1(x_3, y_3) = 0. \tag{7.45}$$

That is, a shape function takes the value unity at its own grid point and zero at the other grid points in an element:

$$N_i(x_j, y_j) = \delta_{ij}. \tag{7.46}$$

Eq. 7.38 gives the shape functions for the linear triangle. An interesting observation is that, if the shape functions are evaluated at the triangle centroid, $N_1 = N_2 = N_3 = 1/3$, since each grid point has equal influence on the centroid. To prove this result, substitute the coordinates of the centroid,

$$\bar{x} = (x_1 + x_2 + x_3)/3, \quad \bar{y} = (y_1 + y_2 + y_3)/3, \tag{7.47}$$

into N_1 in Eq. 7.38, and use Eq. 7.33:

$$N_1(\bar{x}, \bar{y}) = [(x_2y_3 - x_3y_2) + (y_2 - y_3)\bar{x} + (x_3 - x_2)\bar{y}]/(2A) = \frac{1}{3}. \tag{7.48}$$

The preceding discussion assumed an element having three grid points. In general, for an element having r points,

$$\phi(x, y) = N_1(x, y)\phi_1 + N_2(x, y)\phi_2 + \cdots + N_r(x, y)\phi_r = N_i\phi_i. \tag{7.49}$$

This is a convenient way to represent the finite element approximation within an element, since, once the grid point values are known, ϕ can be evaluated anywhere in the element.

We note that, since the shape functions for the 3-point triangle are linear in x and y, the gradients in the x or y directions are constant. Thus, from Eq. 7.37,

$$\frac{\partial\phi}{\partial x} = \frac{\partial N_1}{\partial x}\phi_1 + \frac{\partial N_2}{\partial x}\phi_2 + \frac{\partial N_3}{\partial x}\phi_3 = [(y_2 - y_3)\phi_1 + (y_3 - y_1)\phi_2 + (y_1 - y_2)\phi_3]/(2A) \tag{7.50}$$

$$\frac{\partial\phi}{\partial y} = \frac{\partial N_1}{\partial y}\phi_1 + \frac{\partial N_2}{\partial y}\phi_2 + \frac{\partial N_3}{\partial y}\phi_3 = [(x_3 - x_2)\phi_1 + (x_1 - x_3)\phi_2 + (x_2 - x_1)\phi_3]/(2A). \tag{7.51}$$

A constant gradient within any element implies that many small elements would have to be used wherever ϕ is changing rapidly.

Example. The displacement $u(x)$ of an axial structural member (a pin-jointed truss member) (Fig. 58) is given by

$$u(x) = N_1(x)u_1 + N_2(x)u_2, \tag{7.52}$$

where u_1 and u_2 are the axial displacements at the two end points. For a linear variation of displacement along the length, it follows that N_i must be a linear function which is unity at Point i and zero at the other end. Thus,

$$\begin{cases} N_1(x) = 1 - \dfrac{x}{L}, \\ N_2(x) = \dfrac{x}{L} \end{cases} \tag{7.53}$$

or

$$u(x) = \left(1 - \frac{x}{L}\right) u_1 + \left(\frac{x}{L}\right) u_2. \tag{7.54}$$

7.4 Variational Approach

Consider Poisson's equation in a two-dimensional domain subject to both Dirichlet and Neumann boundary conditions. We consider a slight generalization of the problem of Eq. 7.14:

$$\begin{cases} \dfrac{\partial^2 \phi}{\partial x^2} + \dfrac{\partial^2 \phi}{\partial y^2} + f = 0 \ \text{ in } A, \\ \phi = \phi_0 \ \text{ on } S_1, \\ \dfrac{\partial \phi}{\partial n} + g + h\phi = 0 \ \text{ on } S_2, \end{cases} \tag{7.55}$$

where S_1 and S_2 are curves in 2-D, and f, g, and h may depend on position. The difference between this problem and that of Eq. 7.14 is that the h term has been added to the boundary condition on S_2, where the gradient $\partial\phi/\partial n$ is specified. The h term could arise in a variety of physical situations, including heat transfer due to convection (where the heat flux is proportional to temperature) and free surface flow problems (where the free surface and radiation boundary conditions are both of this type). Assume that the domain has been subdivided into a mesh of triangular finite elements similar to that shown in Fig. 56.

The functional which must be minimized for this boundary value problem is similar to that of Eq. 7.24 except that an additional term must be added to account for the h term in the boundary condition on S_2. It can be shown that the functional which must be minimized for this problem is

$$I(\phi) = \int_A \left\{ \frac{1}{2} \left[\left(\frac{\partial \phi}{\partial x}\right)^2 + \left(\frac{\partial \phi}{\partial y}\right)^2 \right] - f\phi \right\} dA + \int_{S_2} \left(g\phi + \frac{1}{2} h\phi^2 \right) dS, \tag{7.56}$$

where A is the domain.

With a finite element discretization, we do not define a single smooth function ϕ over the entire domain, but instead define ϕ over individual elements. Thus, since I is an integral, it can be evaluated by summing over the elements:

$$I = I_1 + I_2 + I_3 + \cdots = \sum_{e=1}^{E} I_e, \tag{7.57}$$

where E is the number of elements. The variation of I is also computed by summing over the elements:

$$\delta I = \sum_{e=1}^{E} \delta I_e, \tag{7.58}$$

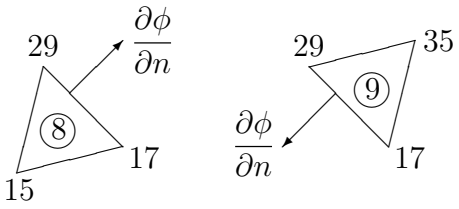


Figure 59: Neumannn Boundary Condition at Internal Boundary.

which must vanish for I to be a minimum.

We thus can consider a typical element. For one element, in index notation,

$$I_e = \int_A \left(\frac{1}{2}\phi_{,k}\phi_{,k} - f\phi\right) dA + \int_{S_2} \left(g\phi + \frac{1}{2}h\phi^2\right) dS. \tag{7.59}$$

The last two terms in this equation are, from Eq. 7.55c, integrals of the form

$$\int_{S_2} \frac{\partial\phi}{\partial n}\phi \, dS.$$

As can be seen from Fig. 59, for two elements which share a common edge, the unknown ϕ is continuous along that edge, and the normal derivative $\partial\phi/\partial n$ is of equal magnitude and opposite sign, so that the individual contributions cancel each other out. Thus, the last two terms in the functional make a nonzero contribution to the functional only if an element abuts S_2 (i.e., if one edge of an element coincides with S_2).

The degrees of freedom which define the function ϕ over the entire domain are the nodal (grid point) values ϕ_i, since, given all the ϕ_i, ϕ is known everywhere using the element shape functions. Therefore, to minimize I, we differentiate with respect to each ϕ_i, and set the result to zero:

$$\frac{\partial I_e}{\partial\phi_i} = \int_A \left[\frac{1}{2}\frac{\partial}{\partial\phi_i}(\phi_{,k})\phi_{,k} + \frac{1}{2}\phi_{,k}\frac{\partial}{\partial\phi_i}(\phi_{,k}) - f\frac{\partial\phi}{\partial\phi_i}\right] dA + \int_{S_2} \left(g\frac{\partial\phi}{\partial\phi_i} + h\phi\frac{\partial\phi}{\partial\phi_i}\right) dS \tag{7.60}$$

$$= \int_A \left[\phi_{,k}\frac{\partial}{\partial\phi_i}(\phi_{,k}) - f\frac{\partial\phi}{\partial\phi_i}\right] dA + \int_{S_2} \left(g\frac{\partial\phi}{\partial\phi_i} + h\phi\frac{\partial\phi}{\partial\phi_i}\right) dS. \tag{7.61}$$

where $\phi = N_j\phi_j$ implies

$$\phi_{,k} = (N_j\phi_j)_{,k} = N_{j,k}\phi_j, \tag{7.62}$$

$$\frac{\partial}{\partial\phi_i}(\phi_{,k}) = \frac{\partial}{\partial\phi_i}(N_{j,k}\phi_j) = N_{j,k}\frac{\partial\phi_j}{\partial\phi_i} = N_{j,k}\delta_{ij} = N_{i,k}, \tag{7.63}$$

and

$$\frac{\partial\phi}{\partial\phi_i} = \frac{\partial}{\partial\phi_i}(N_j\phi_j) = N_j\frac{\partial\phi_j}{\partial\phi_i} = N_j\delta_{ij} = N_i. \tag{7.64}$$

We now substitute these last three equations into Eq. 7.61 to obtain

$$\frac{\partial I_e}{\partial\phi_i} = \int_A (N_{j,k}\phi_j N_{i,k} - fN_i)\, dA + \int_{S_2} (gN_i + hN_j\phi_j N_i)\, dS \tag{7.65}$$

$$= \left(\int_A N_{i,k}N_{j,k}\, dA\right)\phi_j - \left(\int_A fN_i\, dA - \int_{S_2} gN_i\, dS\right) + \left(\int_{S_2} hN_iN_j\, dS\right)\phi_j \tag{7.66}$$

$$= K_{ij}\phi_j - F_i + H_{ij}\phi_j, \tag{7.67}$$

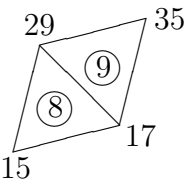


Figure 60: Two Adjacent Finite Elements.

where, for each element,

$$K_{ij}^e = \int_A N_{i,k} N_{j,k} \, dA \tag{7.68}$$

$$F_i^e = \int_A f N_i \, dA - \int_{S_2} g N_i \, dS \tag{7.69}$$

$$H_{ij}^e = \int_{S_2} h N_i N_j \, dS. \tag{7.70}$$

The second term of the "load" $\mathbf{F}$ is present only if Point i is on S_2, and the matrix entries in $\mathbf{H}$ apply only if Points i, j are both on the boundary. The two terms in F correspond to the body force and Neumann boundary condition, respectively.

The superscript e in the preceding equations indicates that we have computed the contribution for one element. We must combine the contributions for all elements. Consider two adjacent elements, as illustrated in Fig. 60. In that figure, the circled numbers are the element labels. From Eq. 7.67, for Element 8, for the special case $h = 0$,

$$\frac{\partial I_8}{\partial \phi_{15}} = K_{15,15}^8 \phi_{15} + K_{15,17}^8 \phi_{17} + K_{15,29}^8 \phi_{29} - F_{15}^8, \tag{7.71}$$

$$\frac{\partial I_8}{\partial \phi_{17}} = K_{17,15}^8 \phi_{15} + K_{17,17}^8 \phi_{17} + K_{17,29}^8 \phi_{29} - F_{17}^8, \tag{7.72}$$

$$\frac{\partial I_8}{\partial \phi_{29}} = K_{29,15}^8 \phi_{15} + K_{29,17}^8 \phi_{17} + K_{29,29}^8 \phi_{29} - F_{29}^8. \tag{7.73}$$

Similarly, for Element 9,

$$\frac{\partial I_9}{\partial \phi_{17}} = K_{17,17}^9 \phi_{17} + K_{17,29}^9 \phi_{29} + K_{17,35}^9 \phi_{35} - F_{17}^9, \tag{7.74}$$

$$\frac{\partial I_9}{\partial \phi_{29}} = K_{29,17}^9 \phi_{17} + K_{29,29}^9 \phi_{29} + K_{29,35}^9 \phi_{35} - F_{29}^9, \tag{7.75}$$

$$\frac{\partial I_9}{\partial \phi_{35}} = K_{35,17}^9 \phi_{17} + K_{35,29}^9 \phi_{29} + K_{35,35}^9 \phi_{35} - F_{35}^9. \tag{7.76}$$

To evaluate

$$\sum_e \frac{\partial I_e}{\partial \phi_i},$$

we sum on e for fixed i. For example, for $i = 17$,

$$\frac{\partial I_8}{\partial \phi_{17}} + \frac{\partial I_9}{\partial \phi_{17}} = K^8_{17,15}\phi_{15} + (K^8_{17,17} + K^9_{17,17})\phi_{17} + (K^8_{17,29} + K^9_{17,29})\phi_{29} + K^9_{17,35}\phi_{35} - F^8_{17} - F^9_{17}. \tag{7.77}$$

This process is then repeated for all grid points and all elements. To minimize the functional, the individual sums are set to zero, resulting in a set of linear algebraic equations of the form

$$\mathbf{K}\boldsymbol{\phi} = \mathbf{F} \tag{7.78}$$

or, for the more general case where $h \neq 0$,

$$(\mathbf{K} + \mathbf{H})\boldsymbol{\phi} = \mathbf{F}, \tag{7.79}$$

where $\boldsymbol{\phi}$ is the vector of unknown nodal values of ϕ, and $\mathbf{F}$ is the vector of forcing functions at the nodes. Because of the historical developments in structural mechanics, $\mathbf{K}$ is sometimes referred to as the *stiffness* matrix. For each element, the contributions to $\mathbf{K}$, $\mathbf{F}$, and $\mathbf{H}$ are

$$K_{ij} = \int_A N_{i,k} N_{j,k}\, dA, \tag{7.80}$$

$$F_i = \int_A f N_i\, dA - \int_{S_2} g N_i\, dS, \tag{7.81}$$

$$H_{ij} = \int_{S_2} h N_i N_j\, dS. \tag{7.82}$$

The sum on k in the first integral can be expanded to yield

$$K_{ij} = \int_A \left(\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y} \right) dA. \tag{7.83}$$

Note that, in three dimensions, Eq. 7.80 still applies, except that the sum extends over the range 1–3, and the integration is over the element volume.

Note also in Eq. 7.81 that, if $g = 0$, there is no contribution to the right-hand side vector $\mathbf{F}$. Thus, to implement the Neumann boundary condition $\partial\phi/\partial n = 0$ at a boundary, the boundary is left free. The zero gradient boundary condition is the *natural* boundary condition for this formulation, since a zero gradient results by default if the boundary is left free. The Dirichlet boundary condition $\phi = \phi_0$ must be explicitly imposed and is referred to as an *essential* boundary condition.

7.5 Matrices for Linear Triangle

Consider the linear three-node triangle, for which, from Eq. 7.39, the shape functions are given by

$$N_i(x, y) = \frac{1}{2A}[(x_j y_k - x_k y_j) + (y_j - y_k)x + (x_k - x_j)y], \tag{7.84}$$

where ijk can be any cyclic permutation of 123. To compute $\mathbf{K}$ from Eq. 7.83, we first compute the derivatives

$$\frac{\partial N_i}{\partial x} = (y_j - y_k)/(2A) = b_i/(2A), \tag{7.85}$$

$$\frac{\partial N_i}{\partial y} = (x_k - x_j)/(2A) = c_i/(2A), \tag{7.86}$$

where b_i and c_i are defined as

$$b_i = y_j - y_k, \quad c_i = x_k - x_j. \tag{7.87}$$

By a cyclic permutation of the indices, we obtain

$$\frac{\partial N_j}{\partial x} = (y_k - y_i)/(2A) = b_j/(2A), \tag{7.88}$$

$$\frac{\partial N_j}{\partial y} = (x_i - x_k)/(2A) = c_j/(2A). \tag{7.89}$$

For the linear triangle, these derivatives are all constant and hence can be removed from the integral to yield

$$K_{ij} = \frac{1}{4A^2}(b_i b_j + c_i c_j)|A|, \tag{7.90}$$

where $|A|$ is the area of the triangular element. Thus, the i, j contribution for one element is

$$K_{ij} = \frac{1}{4|A|}(b_i b_j + c_i c_j), \tag{7.91}$$

where i and j each have the range 123, since there are three grid points in the element. However, b_i and c_i are computed from Eq. 7.87 using the shorthand notation that ijk is a cyclic permutation of 123; that is, $ijk = 123$, $ijk = 231$, or $ijk = 312$. Thus,

$$K_{11} = (b_1^2 + c_1^2)/(4|A|) = [(y_2 - y_3)^2 + (x_3 - x_2)^2]/(4|A|), \tag{7.92}$$

$$K_{22} = (b_2^2 + c_2^2)/(4|A|) = [(y_3 - y_1)^2 + (x_1 - x_3)^2]/(4|A|), \tag{7.93}$$

$$K_{33} = (b_3^2 + c_3^2)/(4|A|) = [(y_1 - y_2)^2 + (x_2 - x_1)^2]/(4|A|), \tag{7.94}$$

$$K_{12} = (b_1 b_2 + c_1 c_2)/(4|A|) = [(y_2 - y_3)(y_3 - y_1) + (x_3 - x_2)(x_1 - x_3)]/(4|A|), \tag{7.95}$$

$$K_{13} = (b_1 b_3 + c_1 c_3)/(4|A|) = [(y_2 - y_3)(y_1 - y_2) + (x_3 - x_2)(x_2 - x_1)]/(4|A|), \tag{7.96}$$

$$K_{23} = (b_2 b_3 + c_2 c_3)/(4|A|) = [(y_3 - y_1)(y_1 - y_2) + (x_1 - x_3)(x_2 - x_1)]/(4|A|). \tag{7.97}$$

Note that $\mathbf{K}$ depends only on *differences* in grid point coordinates. Thus, two elements that are geometrically congruent and differ only by a translation will have the same element matrix.

To compute the right-hand side contributions as given in Eq. 7.81, consider first the contribution of the source term f. From Eq. 7.81, we calculate F_1, which is typical, as

$$F_1 = \int_A f N_1 \, dA = \int_A \frac{f}{2A}[(x_2 y_3 - x_3 y_2) + (y_2 - y_3)x + (x_3 - x_2)y] \, dA. \tag{7.98}$$

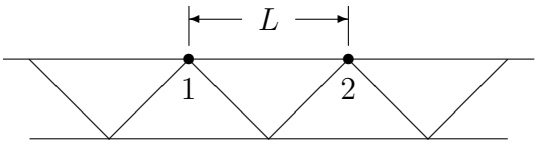


Figure 61: Triangular Mesh at Boundary.

We now consider the special case $f = \hat{f}$ (a constant), and note that

$$\int_A dA = |A|, \quad \int_A x\, dA = \bar{x}\,|A|, \quad \int_A y\, dA = \bar{y}\,|A|, \tag{7.99}$$

where $(\bar{x}, \bar{y})$ is the centroid of the triangle given by Eq. 7.47. Thus,

$$F_1 = \frac{\hat{f}}{2A}\left[(x_2y_3 - x_3y_2) + (y_2 - y_3)\bar{x} + (x_3 - x_2)\bar{y}\right]|A|. \tag{7.100}$$

From Eq. 7.48, the expression in brackets is given by $2A/3$. Hence,

$$F_1 = \frac{1}{3}\hat{f}|A|, \tag{7.101}$$

and similarly

$$F_1 = F_2 = F_3 = \frac{1}{3}\hat{f}|A|. \tag{7.102}$$

That is, for a uniform f, 1/3 of the total element "load" is applied to each grid point. For nonuniform f, the integral could be computed using natural (or area) coordinates for the triangle [7].

It is also of interest to calculate, for the linear triangle, the right-hand side contributions for the second term of Eq. 7.81 for the uniform special case $g = \hat{g}$, where $\hat{g}$ is a constant. For the second term of Eq. 7.81 to contribute, Point i must be on an element edge which lies on the boundary S_2. Since the integration involving g is on the boundary, the only shape functions needed are those which describe the interpolation of ϕ on the boundary. Thus, since the triangular shape functions are linear, the *boundary* shape functions are

$$N_1 = 1 - \frac{x}{L}, \quad N_2 = \frac{x}{L}, \tag{7.103}$$

where the subscripts 1 and 2 refer to the two element edge grid points on the boundary S_2 (Fig. 61), x is a local coordinate along the element edge, and L is the length of the element edge. Thus, for Point i on the boundary,

$$F_i = -\int_{S_2} gN_i\, dS. \tag{7.104}$$

Since the boundary shape functions are given by Eq. 7.103,

$$F_1 = -\hat{g}\int_0^L \left(1 - \frac{x}{L}\right) dx = -\frac{\hat{g}L}{2}, \tag{7.105}$$

$$F_2 = -\hat{g}\int_0^L \left(\frac{x}{L}\right) dx = -\frac{\hat{g}L}{2}. \tag{7.106}$$

Thus, for the two points defining a triangle edge on the boundary S_2,

$$F_1 = F_2 = -\frac{\hat{g}L}{2}. \tag{7.107}$$

That is, for a uniform g, 1/2 of the total element "load" is applied to each grid point.

To calculate $\mathbf{H}$ using Eq. 7.82, we first note that Points i, j must both be on the boundary for this matrix to contribute. Consider some triangular elements adjacent to a boundary, as shown in Fig. 61. Since the integration in Eq. 7.82 is on the boundary, the only shape functions needed are those which describe the interpolation of ϕ on the boundary. Thus, using the boundary shape functions of Eq. 7.103 in Eq. 7.82, for constant $h = \hat{h}$,

$$H_{11} = \hat{h}\int_0^L \left(1 - \frac{x}{L}\right)^2 dx = \frac{\hat{h}L}{3} \tag{7.108}$$

$$H_{22} = \hat{h}\int_0^L \left(\frac{x}{L}\right)^2 dx = \frac{\hat{h}L}{3} \tag{7.109}$$

$$H_{12} = H_{21} = \hat{h}\int_0^L \left(1 - \frac{x}{L}\right)\left(\frac{x}{L}\right) dx = \frac{\hat{h}L}{6}. \tag{7.110}$$

Hence, for an edge of a linear 3-node triangle,

$$\mathbf{H} = \frac{\hat{h}L}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}. \tag{7.111}$$

7.6 Interpretation of Functional

Now that the variational problem has been solved (i.e., the functional I has been minimized), we can evaluate I. We recall from Eq. 7.59 (with $h = 0$) that, for the two-dimensional Poisson's equation,

$$I(\phi) = \int_A \left(\frac{1}{2}\phi_{,k}\phi_{,k} - f\phi\right) dA + \int_{S_2} g\phi\, dS, \tag{7.112}$$

where

$$\phi = N_i\phi_i. \tag{7.113}$$

Since ϕ_i is independent of position, it follows that $\phi_{,k} = N_{i,k}\phi_i$ and

$$I(\phi) = \int_A \left(\frac{1}{2}N_{i,k}\phi_i N_{j,k}\phi_j - fN_i\phi_i\right) dA + \int_{S_2} gN_i\phi_i\, dS, \tag{7.114}$$

$$= \frac{1}{2}\phi_i\left(\int_A N_{i,k}N_{j,k}\, dA\right)\phi_j - \left(\int_A fN_i\, dA - \int_{S_2} gN_i\, dS\right)\phi_i \tag{7.115}$$

$$= \frac{1}{2}\phi_i K_{ij}\phi_j - F_i\phi_i \quad \text{(index notation)} \tag{7.116}$$

$$= \frac{1}{2}\boldsymbol{\phi}^T\mathbf{K}\boldsymbol{\phi} - \boldsymbol{\phi}^T\mathbf{F} \quad \text{(matrix notation)} \tag{7.117}$$

$$= \frac{1}{2}\boldsymbol{\phi}\cdot\mathbf{K}\boldsymbol{\phi} - \boldsymbol{\phi}\cdot\mathbf{F} \quad \text{(vector notation)}, \tag{7.118}$$

where the last result has been written in index, matrix, and vector notations. The first term for I is a quadratic form.

In solid mechanics, I corresponds to the total potential energy. The first term is the strain energy, and the second term is the potential of the applied loads. Since strain energy is zero only for zero strain (corresponding to rigid body motion), it follows that the stiffness matrix $\mathbf{K}$ is positive semi-definite. For well-posed problems (which allow no rigid body motion), $\mathbf{K}$ is positive definite. (By definition, a matrix $\mathbf{K}$ is positive definite if $\boldsymbol{\phi}^T\mathbf{K}\boldsymbol{\phi} > 0$ for all $\boldsymbol{\phi} \neq \mathbf{0}$.) Three consequences of positive-definiteness are

1. All eigenvalues of $\mathbf{K}$ are positive.
2. The matrix is non-singular.
3. Gaussian elimination can be performed without pivoting.

7.7 Stiffness in Elasticity in Terms of Shape Functions

In §4.10 (p. 45), the Principle of Virtual Work was used to obtain the element stiffness matrix for an elastic finite element as (Eq. 4.67)

$$\mathbf{K} = \int_V \mathbf{C}^T\mathbf{D}\mathbf{C}\, dV, \tag{7.119}$$

where $\mathbf{D}$ is the symmetric matrix of material constants relating stress and strain, and $\mathbf{C}$ is the matrix converting grid point displacements to strain:

$$\boldsymbol{\varepsilon} = \mathbf{C}\mathbf{u}. \tag{7.120}$$

For the constant strain triangle (CST) in 2-D, for example, the fundamental unknowns u and v (the x and y components of displacement) can both be expressed in terms of the three shape functions defined in Eq. 7.38:

$$u = N_i u_i, \quad v = N_i v_i, \tag{7.121}$$

where the summation convention is used. In general, in 2-D, the strains can be expressed as

$$\begin{aligned}\boldsymbol{\varepsilon} &= \left\{ \begin{array}{c} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{array} \right\} = \left\{ \begin{array}{c} u_{,x} \\ v_{,y} \\ u_{,y} + v_{,x} \end{array} \right\} = \left\{ \begin{array}{c} N_{i,x}u_i \\ N_{i,y}v_i \\ N_{i,y}u_i + N_{i,x}v_i \end{array} \right\} \\ &= \left[\begin{array}{cccccc} N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 \\ 0 & N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} \\ N_{1,y} & N_{1,x} & N_{2,y} & N_{2,x} & N_{3,y} & N_{3,x} \end{array} \right] \left\{ \begin{array}{c} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \end{array} \right\}. \end{aligned} \tag{7.122}$$

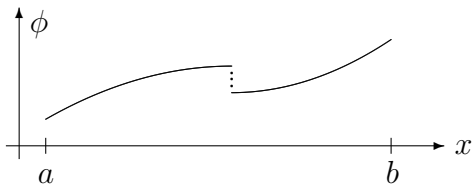


Figure 62: Discontinuous Function.

Thus, from Eq. 7.120, $\mathbf{C}$ in Eq. 7.119 is a matrix of shape function derivatives:

$$\mathbf{C} = \begin{bmatrix} N_{1,x} & 0 & N_{2,x} & 0 & N_{3,x} & 0 \\ 0 & N_{1,y} & 0 & N_{2,y} & 0 & N_{3,y} \\ N_{1,y} & N_{1,x} & N_{2,y} & N_{2,x} & N_{3,y} & N_{3,x} \end{bmatrix}. \tag{7.123}$$

In general, $\mathbf{C}$ would have as many rows as there are independent components of strain (3 in 2-D and 6 in 3-D) and as many columns as there are DOF in the element.

7.8 Element Compatibility

In the variational approach to the finite element method, an integral was minimized. It was also assumed that the integral evaluated over some domain was equal to the sum of the integrals over the elements. We wish to investigate briefly the conditions which must hold for this assumption to be valid. That is, what condition is necessary for the integral over a domain to be equal to the sum of the integrals over the elements?

Consider the one-dimensional integral

$$I = \int_a^b \phi(x)\, dx. \tag{7.124}$$

For the integral I to be well-defined, simple jump discontinuities in ϕ are allowed, as illustrated in Fig. 62. Singularities in ϕ, on the other hand, will not be allowed, since some singularities cannot be integrated. Thus, we conclude that, for any functional of interest in finite element analysis, the integrand may be discontinuous, but we do not allow singularities in the integrand.

Now consider the one-dimensional integral

$$I = \int_a^b \frac{d\phi(x)}{dx}\, dx. \tag{7.125}$$

Since the integrand $d\phi(x)/dx$ may be discontinuous, $\phi(x)$ must be continuous, but with kinks (slope discontinuities). In the integral

$$I = \int_a^b \frac{d^2\phi(x)}{dx^2}\, dx, \tag{7.126}$$

the integrand ϕ'' may have simple discontinuities, in which case ϕ' is continuous with kinks, and ϕ is smooth (i.e., the slope is continuous).

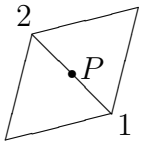


Figure 63: Compatibility at an Element Boundary.

Thus, we conclude that the smoothness required of ϕ depends on the highest order derivative of ϕ appearing in the integrand. If ϕ'' is in the integrand, ϕ' must be continuous. If ϕ' is in the integrand, ϕ must be continuous. Therefore, in general, we conclude that, at element interfaces, the field variable ϕ and any of its partial derivatives up to one order less than the highest order derivative appearing in $I(\phi)$ must be continuous. This requirement is referred to as the *compatibility requirement* or the *conforming requirement*.

For example, consider the Poisson equation in 2-D,

$$\frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2} + f = 0, \tag{7.127}$$

for which the functional to be minimized is

$$I(\phi) = \int_A \left\{ \frac{1}{2}\left[\left(\frac{\partial\phi}{\partial x}\right)^2 + \left(\frac{\partial\phi}{\partial y}\right)^2 \right] - f\phi \right\} dA. \tag{7.128}$$

Since the highest derivative in I is first order, we conclude that, for I to be well-defined, ϕ must be continuous in the finite element approximation.

For the 3-node triangular element already formulated, the shape function is linear, which implies that, in Fig. 63, given ϕ_1 and ϕ_2, ϕ varies linearly between ϕ_1 and ϕ_2 along the line 1–2. That is, ϕ is the same at the mid-point P for both elements; otherwise, $I(\phi)$ might be infinite, since there could be a gap or overlap in the model along the line 1–2.

Note also that the Poisson equation is a second order equation, but ϕ need only be continuous in the finite element approximation. That is, the first derivatives $\phi_{,x}$ and $\phi_{,y}$ may have simple discontinuities, and the second derivatives $\phi_{,xx}$ and $\phi_{,yy}$ that appear in the partial differential equation may not even exist at the element interfaces. Thus, one of the strengths of the variational approach is that $I(\phi)$ involves derivatives of lower order than in the original PDE.

In elasticity, the functional $I(\phi)$ has the physical interpretation of total potential energy, including strain energy. A nonconforming element would result in a discontinuous displacement at the element boundaries (i.e., a gap or overlap), which would correspond to infinite strain energy. However, note that having displacement continuous implies that the displacement gradients (which are proportional to the stresses) are discontinuous at the element boundaries. This property is one of the fundamental characteristics of an approximate numerical solution. If all quantities of interest (e.g., displacements and stresses) were continuous, the solution would be an exact solution rather than an approximate solution. Thus, the approximation inherent in a displacement-based finite element method is that the displacements are continuous, and the stresses (displacement gradients) are discontinuous at element boundaries. In fluid mechanics, a discontinuous ϕ (which is not allowed) corresponds to a singularity in the velocity field.

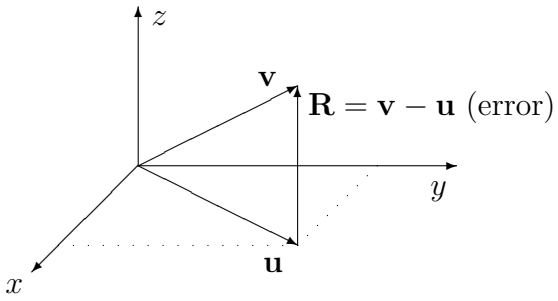


Figure 64: A Vector Analogy for Galerkin's Method.

7.9 Method of Weighted Residuals (Galerkin's Method)

Here we discuss an alternative to the use of a variational principle when the functional is either unknown or does not exist (e.g., nonlinear equations).

We consider the Poisson equation

$$\nabla^2\phi + f = 0. \tag{7.129}$$

For an approximate solution $\tilde{\phi}$,

$$\nabla^2\tilde{\phi} + f = R \neq 0, \tag{7.130}$$

where R is referred to as the *residual* or error. The best approximate solution will be one which in some sense minimizes R at all points of the domain. We note that, if $R = 0$ in the domain,

$$\int_V RW\, dV = 0, \tag{7.131}$$

where W is any function of the spatial coordinates. W is referred to as a *weighting* function. With n DOF in the domain, n functions W can be chosen:

$$\int_V RW_i\, dV = 0, \quad i = 1, 2, \ldots, n. \tag{7.132}$$

This approach is called the *method of weighted residuals*. Various choices of W_i are possible. When $W_i = N_i$ (the shape functions), the process is called *Galerkin's method*.

The motivation for using shape functions N_i as the weighting functions is that we want the residual (the error) orthogonal to the shape functions. In the finite element approximation, we are trying to approximate an infinite DOF problem (the PDE) with a finite number of DOF (the finite element model). Consider an analogous problem in vector analysis, where we want to approximate a vector $\mathbf{v}$ in 3-D with another vector in 2-D. That is, we are attempting to approximate $\mathbf{v}$ with a lesser number of DOF, as shown in Fig. 64. The "best" 2-D approximation to the 3-D vector $\mathbf{v}$ is the projection $\mathbf{u}$ in the plane. The error in this approximation is

$$\mathbf{R} = \mathbf{v} - \mathbf{u}, \tag{7.133}$$

which is orthogonal to the xy-plane. That is, the error $\mathbf{R}$ is orthogonal to the basis vectors $\mathbf{e}_x$ and $\mathbf{e}_y$ (the vectors used to approximate $\mathbf{v}$):

$$\mathbf{R} \cdot \mathbf{e}_x = 0 \quad \text{and} \quad \mathbf{R} \cdot \mathbf{e}_y = 0. \tag{7.134}$$

In the finite element problem, the approximating functions are the shape functions N_i. The counterpart to the dot product is the integral

$$\int_V R N_i \, dV = 0. \tag{7.135}$$

That is, the residual R is orthogonal to its approximating functions, the shape functions.

The integral in Eq. 7.135 must hold over the entire domain V or any portion of the domain, e.g., an element. Thus, for Poisson's equation, for one element,

$$0 = \int_V (\nabla^2 \phi + f) N_i \, dV \tag{7.136}$$

$$= \int_V (\phi_{,kk} + f) N_i \, dV \tag{7.137}$$

$$= \int_V [(\phi_{,k} N_i)_{,k} - \phi_{,k} N_{i,k}] \, dV + \int_V f N_i \, dV. \tag{7.138}$$

The first term is converted to a surface integral using the divergence theorem, Eq. 7.10, to obtain

$$0 = \oint_S \phi_{,k} N_i n_k \, dS - \int_V \phi_{,k} N_{i,k} \, dV + \int_V f N_i \, dV, \tag{7.139}$$

where

$$\phi_{,k} n_k = \nabla \phi \cdot \mathbf{n} = \frac{\partial \phi}{\partial n} \tag{7.140}$$

and

$$\phi_{,k} = (N_j \phi_j)_{,k} = N_{j,k} \phi_j. \tag{7.141}$$

Hence, for each i,

$$0 = \oint_S \frac{\partial \phi}{\partial n} N_i \, dS - \int_V N_{j,k} \phi_j N_{i,k} \, dV + \int_V f N_i \, dV \tag{7.142}$$

$$= -\left(\int_V N_{i,k} N_{j,k} \, dV \right) \phi_j + \left(\int_V f N_i \, dV + \oint_S \frac{\partial \phi}{\partial n} N_i \, dS \right) \tag{7.143}$$

$$= -K_{ij} \phi_j + F_i. \tag{7.144}$$

Thus, in matrix notation,

$$\mathbf{K}\boldsymbol{\phi} = \mathbf{F}, \tag{7.145}$$

where

$$K_{ij} = \int_V N_{i,k} N_{j,k} \, dV \tag{7.146}$$

$$F_i = \int_V f N_i \, dV + \oint_S \frac{\partial \phi}{\partial n} N_i \, dS. \tag{7.147}$$

From Eq. 7.55, $\partial\phi/\partial n$ is specified on S_2 and unknown *a priori* on S_1, where $\phi = \phi_0$ is specified. On S_1, $\partial\phi/\partial n$ is the "reaction" to the specified ϕ. At points where ϕ is specified, the Dirichlet boundary conditions are handled like displacement boundary conditions in structural problems.

Galerkin's method thus results in algebraic equations identical to those derived from a variational principle. However, Galerkin's method is more general, since sometimes a variational principle may not exist for a given problem. When a principle does exist, the two approaches yield the same results. When the variational principle does not exist or is unknown, Galerkin's method can still be used to derive a finite element model.

8 Potential Fluid Flow With Finite Elements

In potential flow, the fluid is assumed to be inviscid and incompressible. Since there are no shearing stresses in the fluid, the fluid slips tangentially along boundaries. This mathematical model of fluid behavior is useful for some situations.

Define a velocity potential ϕ such that velocity $\mathbf{v} = \nabla\phi$. That is, in 3-D,

$$v_x = \frac{\partial\phi}{\partial x}, \quad v_y = \frac{\partial\phi}{\partial y}, \quad v_z = \frac{\partial\phi}{\partial z}. \tag{8.1}$$

It can be shown that, within the domain occupied by the fluid,

$$\nabla^2\phi = 0. \tag{8.2}$$

Various boundary conditions are of interest. At a fixed boundary, where the normal velocity vanishes,

$$v_n = \mathbf{v}\cdot\mathbf{n} = \nabla\phi\cdot\mathbf{n} = \frac{\partial\phi}{\partial n} = 0, \tag{8.3}$$

where $\mathbf{n}$ is the unit outward normal on the boundary. On a boundary where the velocity is specified,

$$\frac{\partial\phi}{\partial n} = \hat{v}_n. \tag{8.4}$$

We will see later in the discussion of symmetry that, at a plane of symmetry for the potential ϕ,

$$\frac{\partial\phi}{\partial n} = 0, \tag{8.5}$$

where $\mathbf{n}$ is the unit normal to the plane. At a plane of antisymmetry, $\phi = 0$.

The boundary value problem for flow around a solid body is illustrated in Fig. 65. In this example, far away from the body, where the velocity is known,

$$v_x = \frac{\partial\phi}{\partial x} = v_\infty, \tag{8.6}$$

which is specified.

As is, the problem posed in Fig. 65 is not a well-posed problem, because only conditions on the derivative $\partial\phi/\partial n$ are specified. Thus, for any solution ϕ, $\phi + c$ is also a solution for any constant c. Therefore, for uniqueness, we must specify ϕ somewhere in the domain.

Thus, the potential flow boundary value problem is that the velocity potential ϕ satisfies

$$\begin{cases} \nabla^2\phi = 0 \text{ in } V \\ \phi = \hat{\phi} \text{ on } S_1 \\ \dfrac{\partial\phi}{\partial n} = \hat{v}_n \text{ on } S_2, \end{cases} \tag{8.7}$$

where $\mathbf{v} = \nabla\phi$ in V.

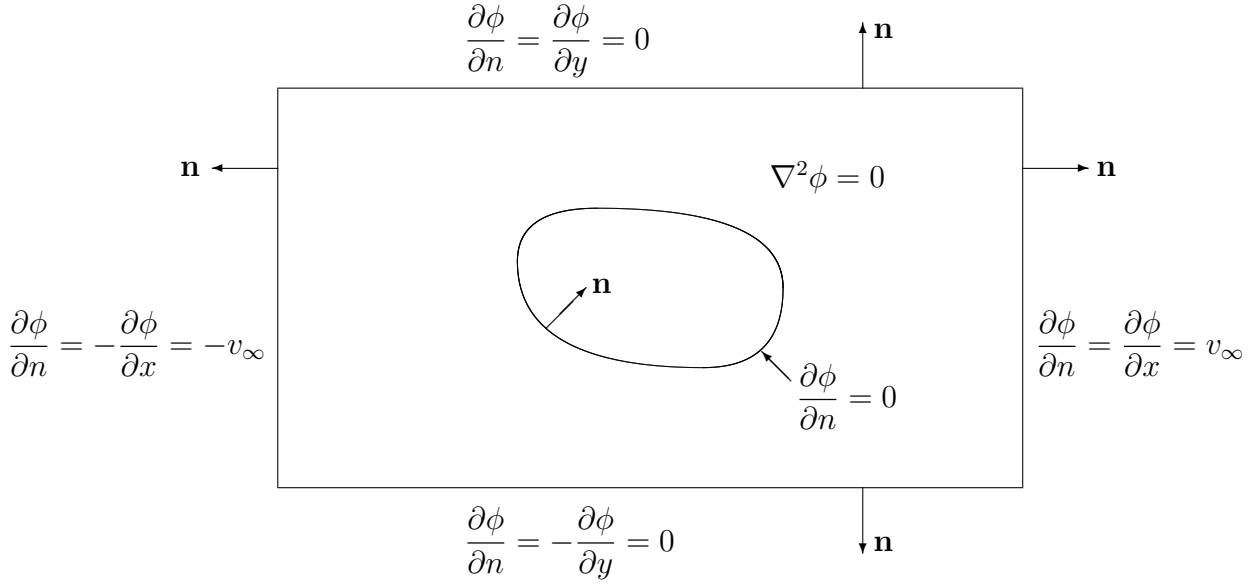


Figure 65: Potential Flow Around Solid Body.

8.1 Finite Element Model

A finite element model of the potential flow problem results in the equation

$$\mathbf{K}\boldsymbol{\phi} = \mathbf{F}, \tag{8.8}$$

where the contributions to $\mathbf{K}$ and $\mathbf{F}$ for each element are

$$K_{ij} = \int_A N_{i,k} N_{j,k}\, dA = \int_A \left(\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y} \right) dA, \tag{8.9}$$

$$F_i = \int_{S_2} \hat{v}_n N_i\, dS, \tag{8.10}$$

where $\hat{v}_n$ is the specified velocity on S_2 in the outward normal direction, and N_i is the shape function for the ith grid point in the element.

Once the velocity potential ϕ is known, the pressure can be found using the steady-state Bernoulli equation

$$\frac{1}{2}v^2 + gy + \frac{p}{\rho} = c = \text{constant}, \tag{8.11}$$

where v is the velocity magnitude given by

$$v^2 = \left(\frac{\partial\phi}{\partial x}\right)^2 + \left(\frac{\partial\phi}{\partial y}\right)^2, \tag{8.12}$$

gy is the (frequently ignored) body force potential, g is the acceleration due to gravity, y is the height above some reference plane, p is pressure, and ρ is the fluid density. The constant c is evaluated using a location where v is known (e.g., v_∞). For example, at infinity, if we ignore gy and pick $p = 0$ (ambient),

$$c = \frac{1}{2}v_\infty^2, \tag{8.13}$$

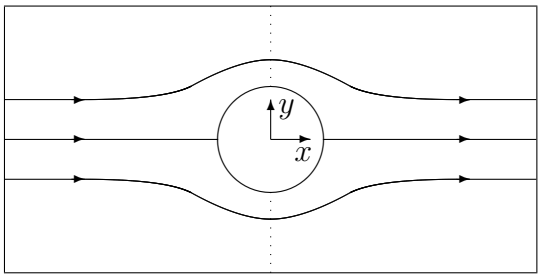


Figure 66: Streamlines Around Circular Cylinder.

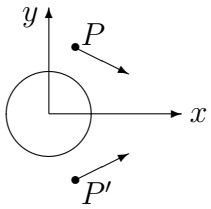


Figure 67: Symmetry With Respect to $y = 0$.

and

$$\frac{p}{\rho} + \frac{1}{2}v^2 = \frac{1}{2}v_\infty^2. \tag{8.14}$$

8.2 Application of Symmetry

Consider the 2-D potential flow around a circular cylinder, as shown in Fig. 66. The velocity field is symmetric with respect to $y = 0$. For example, in Fig. 67, the velocity vectors at P and its image P' are mirror images of each other. As P and P' get close to the axis $y = 0$, the velocities at P and P' must converge to each other, since P and P' are the same point in the plane $y = 0$. Thus,

$$v_y = \frac{\partial \phi}{\partial y} = 0 \quad \text{for } y = 0. \tag{8.15}$$

The y-direction in this case is the normal to the symmetry plane $y = 0$. Thus, in general, we conclude that, for points in a symmetry plane with normal $\mathbf{n}$,

$$\frac{\partial \phi}{\partial n} = 0. \tag{8.16}$$

Note from Fig. 65 that the specified normal velocities at the two x extremes are of opposite signs. Thus, from Eq. 8.10, the right-hand side "loads" in Eq. 8.8 are equal in magnitude and opposite in sign for the left and right boundaries, and the velocity field is antisymmetric with respect to the plane $x = 0$, as shown in Fig. 68. That is, the velocity vectors at P and P' can be transformed into each other by a reflection and a negation of sign. Thus,

$$(v_x)_P = (v_x)_{P'}. \tag{8.17}$$

If we change the direction of flow (i.e., make it right to left), then

$$(\phi_P)_{\text{flow right}} = (\phi_{P'})_{\text{flow left}}. \tag{8.18}$$

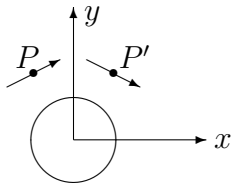


Figure 68: Antisymmetry With Respect to $x = 0$.

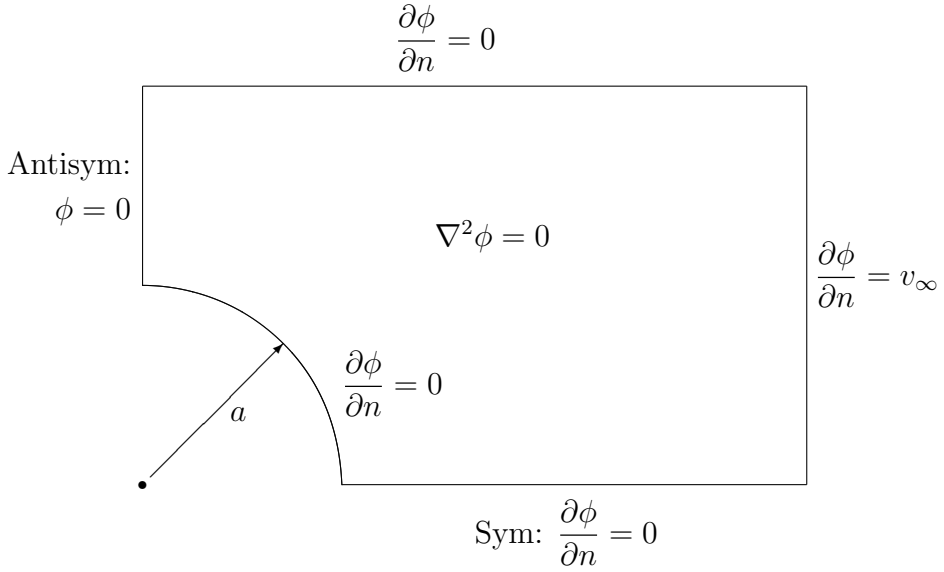


Figure 69: Boundary Value Problem for Flow Around Circular Cylinder.

However, changing the direction of flow also means that $\mathbf{F}$ in Eq. 8.8 becomes $-\mathbf{F}$, since the only nonzero contributions to $\mathbf{F}$ occur at the left and right boundaries. However, if the sign of $\mathbf{F}$ changes, the sign of the solution also changes, i.e.,

$$(\phi_{P'})_{\text{flow right}} = -(\phi_{P'})_{\text{flow left}}. \tag{8.19}$$

Combining the last two equations yields

$$(\phi_P)_{\text{flow right}} = -(\phi_{P'})_{\text{flow right}}. \tag{8.20}$$

If we now let P and P' converge to the plane $x = 0$ and become the same point, we obtain

$$(\phi)_{x=0} = -(\phi)_{x=0} \tag{8.21}$$

or $(\phi)_{x=0} = 0$. Thus, in general, we conclude that, for points in a symmetry plane with normal $\mathbf{n}$ for which the solution is antisymmetric, $\phi = 0$. The key to recognizing antisymmetry is to have a symmetric geometry with an antisymmetric "loading" (RHS).

For example, for 2-D potential flow around a circular cylinder, Fig. 66, the boundary value problem is shown in Fig. 69. This problem has two planes of geometric symmetry, $y = 0$ and $x = 0$, and can be solved using a one-quarter model. Since ϕ is specified on $x = 0$, this problem is well-posed. The boundary condition $\partial\phi/\partial n = 0$ is the natural boundary condition (the condition that results if the boundary is left free).

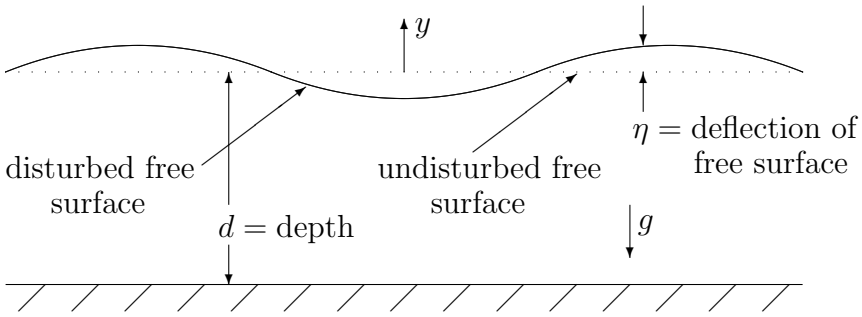


Figure 70: The Free Surface Problem.

8.3 Free Surface Flows

Consider an inviscid, incompressible fluid in an irrotational flow field with a free surface, as shown in Fig. 70. The equations of motion and continuity reduce to

$$\nabla^2 \phi = 0, \tag{8.22}$$

where ϕ is the velocity potential, and the velocity is

$$\mathbf{v} = \nabla \phi. \tag{8.23}$$

The pressure p can be determined from the time-dependent Bernoulli equation

$$-\frac{p}{\rho} = \frac{\partial \phi}{\partial t} + \frac{1}{2} v^2 + gy, \tag{8.24}$$

where ρ is the fluid density, g is the acceleration due to gravity, and y is the vertical coordinate.

If we let η denote the deflection of the free surface, the vertical velocity on the free surface is

$$\frac{\partial \phi}{\partial y} = \frac{\partial \eta}{\partial t} \quad \text{on } y = 0. \tag{8.25}$$

If we assume small wave height (i.e., η is small compared to the depth d), the velocity v on the free surface is also small, and we can ignore the velocity term in Eq. 8.24. If we also take the pressure $p = 0$ on the free surface, Bernoulli's equation implies

$$\frac{\partial \phi}{\partial t} + g\eta = 0 \quad \text{on } y = 0. \tag{8.26}$$

This equation can be viewed as an equation for the surface elevation η given ϕ. We can then eliminate η from the last two equations by differentiating Eq. 8.26:

$$0 = \frac{\partial^2 \phi}{\partial t^2} + g\frac{\partial \eta}{\partial t} = \frac{\partial^2 \phi}{\partial t^2} + g\frac{\partial \phi}{\partial y}. \tag{8.27}$$

Hence, on the free surface $y = 0$,

$$\frac{\partial \phi}{\partial y} = -\frac{1}{g}\frac{\partial^2 \phi}{\partial t^2}. \tag{8.28}$$

This equation is referred to as the *linearized free surface boundary condition.*

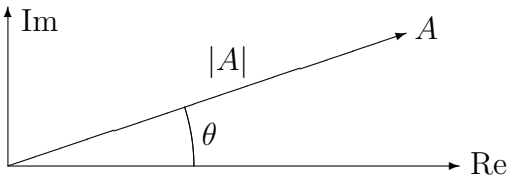


Figure 71: The Complex Amplitude.

8.4 Use of Complex Numbers and Phasors in Wave Problems

The wave maker problem considered in the next section will involve a forcing function which is sinusoidal in time (i.e., time-harmonic). It is common in engineering analysis to represent time-harmonic signals using complex numbers, since amplitude and phase information can be included in a single complex number. Such an approach is used with A.C. circuits, steady-state acoustics, and mechanical vibrations.

Consider a sine wave $\phi(t)$ of amplitude $\hat{A}$, circular frequency ω, and phase θ:

$$\phi(t) = \hat{A}\cos(\omega t + \theta), \tag{8.29}$$

where all quantities in this equation are real, and $\hat{A}$ can be taken as positive. Using complex notation,

$$\phi(t) = \mathrm{Re}\left[\hat{A}e^{i(\omega t+\theta)}\right] = \mathrm{Re}\left[\left(\hat{A}e^{i\theta}\right)e^{i\omega t}\right], \tag{8.30}$$

where $i = \sqrt{-1}$. If we define the *complex amplitude*

$$A = \hat{A}e^{i\theta}, \tag{8.31}$$

then

$$\phi(t) = \mathrm{Re}\left(Ae^{i\omega t}\right), \tag{8.32}$$

where the magnitude of the complex amplitude is given by

$$|A| = \left|\hat{A}e^{i\theta}\right| = \left|\hat{A}\right|\left|e^{i\theta}\right| = \hat{A}, \tag{8.33}$$

which is the actual amplitude, and

$$\arg(A) = \theta, \tag{8.34}$$

which is the actual phase angle. The complex amplitude A is thus a complex number which embodies both the amplitude and the phase of the sinusoidal signal, as shown in Fig. 71. The directed vector in the complex plane is called a *phasor* by electrical engineers.

It is common practice, when dealing with these sinsusoidal functions, to drop the "Re", and agree that it is only the real part which is of interest. Thus, we write

$$\phi(t) = Ae^{i\omega t} \tag{8.35}$$

with the understanding that it is the real part of this signal which is of interest. In this equation, A is the complex amplitude.

Two sinusoids of the same frequency add just like vectors in geometry. For example, consider the sum

$$\hat{A}_1\cos(\omega t + \theta_1) + \hat{A}_2\cos(\omega t + \theta_2).$$

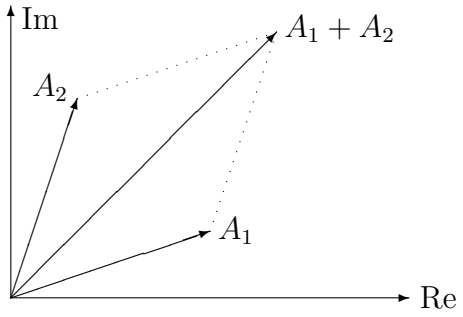


Figure 72: Phasor Addition.

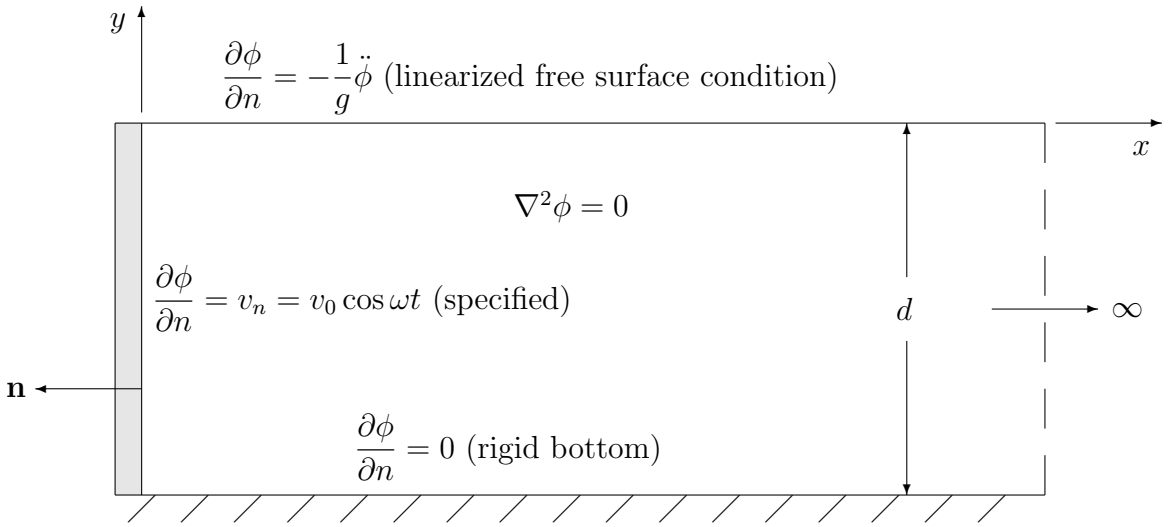


Figure 73: 2-D Wave Maker: Time Domain.

In terms of complex arithmetic,

$$A_1 e^{i\omega t} + A_2 e^{i\omega t} = (A_1 + A_2) e^{i\omega t}, \tag{8.36}$$

where A_1 and A_2 are complex amplitudes given by

$$A_1 = \hat{A}_1 e^{i\theta_1}, \quad A_2 = \hat{A}_2 e^{i\theta_2}. \tag{8.37}$$

This addition, referred to as *phasor addition*, is illustrated in Fig. 72.

8.5 2-D Wave Maker

Consider a semi-infinite body of water with a wall oscillating in simple harmonic motion, as shown in Fig. 73. In the time domain, the problem is

$$\begin{cases} \nabla^2\phi = 0 \\ \dfrac{\partial\phi}{\partial n} = v_0 \cos\omega t \quad \text{on } x = 0 \\ \dfrac{\partial\phi}{\partial n} = 0 \quad \text{on } y = -d \\ \dfrac{\partial\phi}{\partial n} = -\dfrac{1}{g}\ddot{\phi} \quad \text{on } y = 0, \end{cases} \tag{8.38}$$

where dots denote differentiation with respect to time, and an additional boundary condition is needed for large x at the location where the model is terminated. For this problem, the excitation frequency ω is specified. The solution of this problem is a function $\phi(x, y, t)$.

The forcing function is the oscillating wall. We first write Eq. 8.38b in the form

$$\frac{\partial\phi}{\partial n} = v_0 \cos\omega t = \mathrm{Re}\left(v_0 e^{i\omega t}\right), \tag{8.39}$$

where $i = \sqrt{-1}$, and v_0 is real. We therefore look for solutions in the form

$$\phi(x, y, t) = \phi_0(x, y)e^{i\omega t}, \tag{8.40}$$

where $\phi_0(x, y)$ is the complex amplitude. Eq. 8.38 then becomes

$$\begin{cases} \nabla^2\phi_0 = 0 \\ \dfrac{\partial\phi_0}{\partial n} = v_0 \quad \text{on } x = 0 \\ \dfrac{\partial\phi_0}{\partial n} = 0 \quad \text{on } y = -d \\ \dfrac{\partial\phi_0}{\partial n} = \dfrac{\omega^2}{g}\phi_0 \quad \text{on } y = 0. \end{cases} \tag{8.41}$$

It can be shown that, for large x,

$$\frac{\partial\phi_0}{\partial x} = -i\alpha\phi_0, \tag{8.42}$$

where α is the positive solution of

$$\frac{\omega^2}{g} = \alpha \tanh(\alpha d), \tag{8.43}$$

and ω is the fixed excitation frequency. The graphical solution of this equation is shown in Fig. 74.

Thus, for a finite element solution to the 2-D wave maker problem, we truncate the domain "sufficiently far" from the wall, and impose a boundary condition, Eq. 8.42, referred to as a *radiation boundary condition* which accounts approximately for the fact that the

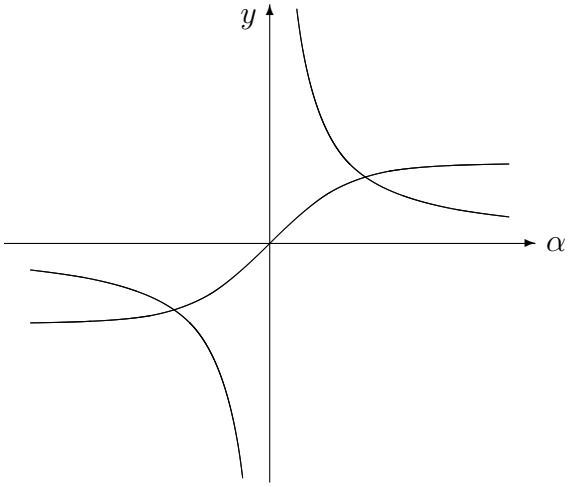


Figure 74: Graphical Solution of $\omega^2/\alpha = g\tanh(\alpha d)$.

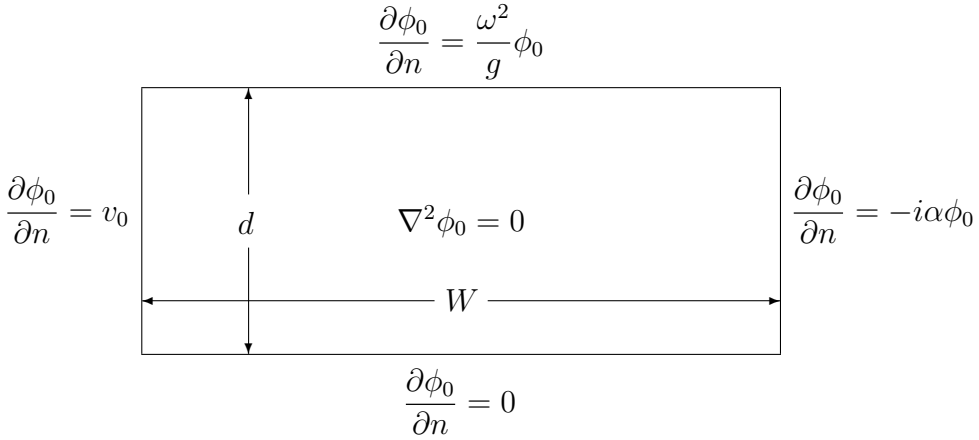


Figure 75: 2-D Wave Maker: Frequency Domain.

fluid extends to infinity. If the radiation boundary is located at $x = W$, the boundary value problem in the frequency domain becomes

$$\begin{cases} \nabla^2\phi_0 = 0 \\ \dfrac{\partial\phi_0}{\partial n} = v_0 \ \text{ on } x = 0 \ \text{ (oscillating wall)} \\ \dfrac{\partial\phi_0}{\partial n} = 0 \ \text{ on } y = -d \ \text{ (rigid bottom)} \\ \dfrac{\partial\phi_0}{\partial n} = \dfrac{\omega^2}{g}\phi_0 \ \text{ on } y = 0 \ \text{ (linearized free surface condition)} \\ \dfrac{\partial\phi_0}{\partial x} = -i\alpha\phi_0 \ \text{ on } x = W \ \text{ (radiation condition),} \end{cases} \tag{8.44}$$

as summarized in Fig. 75. Note the similarity between this radiation condition and the nonreflecting boundary condition for the Helmholtz equation, Eq. 3.52.

8.6 Linear Triangle Matrices for 2-D Wave Maker Problem

The boundary value problem defined in Eq. 8.44 has two boundaries where $\partial\phi/\partial n$ is specified and two boundaries on which $\partial\phi/\partial n$ is proportional to the unknown ϕ. Thus, this boundary value problem is a special case of Eq. 7.55 (p. 73), so that the finite element formulas derived in §7.5 are all applicable. Note that the function g appearing in Eq. 7.55 is not the acceleration due to gravity appearing in the formulation of the free surface flow problem.

The matrix system for the wave maker problem is therefore, from Eq. 7.79,

$$(\mathbf{K} + \mathbf{H})\boldsymbol{\phi} = \mathbf{F}, \tag{8.45}$$

where, for each element, $\mathbf{K}$, $\mathbf{H}$, and $\mathbf{F}$ are given by Eqs. 7.80–7.82. Thus, from Eq. 7.91,

$$K_{ij} = \frac{1}{4|A|}(b_i b_j + c_i c_j), \tag{8.46}$$

where i and j each have the range 123, and

$$b_i = y_j - y_k, \quad c_i = x_k - x_j. \tag{8.47}$$

For b_i and c_i, the symbols ijk refer to the three nodes 123 in a cyclic permutation. For example, if $j = 1$, then $k = 2$ and $i = 3$. Thus,

$$K_{11} = (b_1^2 + c_1^2)/(4|A|), \tag{8.48}$$
$$K_{22} = (b_2^2 + c_2^2)/(4|A|), \tag{8.49}$$
$$K_{33} = (b_3^2 + c_3^2)/(4|A|), \tag{8.50}$$
$$K_{12} = K_{21} = (b_1 b_2 + c_1 c_2)/(4|A|), \tag{8.51}$$
$$K_{13} = K_{31} = (b_1 b_3 + c_1 c_3)/(4|A|), \tag{8.52}$$
$$K_{23} = K_{32} = (b_2 b_3 + c_2 c_3)/(4|A|). \tag{8.53}$$

$\mathbf{H}$ is calculated using Eq. 7.111, where $\hat{h} = -\omega^2/g$ on the free surface, and $\hat{h} = i\alpha$ on the radiation boundary. Thus, for triangular elements adjacent to the free surface,

$$\mathbf{H}_{\mathrm{FS}} = -\frac{\omega^2 L}{6g}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \tag{8.54}$$

for each free surface edge. For elements adjacent to the radiation boundary,

$$\mathbf{H}_{\mathrm{RB}} = \frac{i\alpha L}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \tag{8.55}$$

for each radiation boundary edge. Note that, since $\mathbf{H}$ is purely imaginary on the radiation boundary, the coefficient matrix $\mathbf{K}+\mathbf{H}$ in Eq. 8.45 is complex, and the solution $\boldsymbol{\phi}$ is complex. The solution of free surface problems thus requires either the use of complex arithmetic or separating the matrix system into real and imaginary parts.

The right-hand side $\mathbf{F}$ is calculated using Eq. 7.107, where $\hat{g} = -v_0$ on the oscillating wall. Thus, for two points on an element edge on the oscillating wall,

$$F_1 = F_2 = \frac{v_0 L}{2}. \tag{8.56}$$

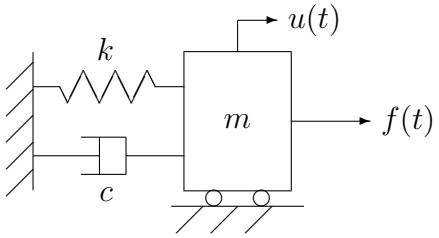


Figure 76: Single DOF Mass-Spring-Dashpot System.

The solution vector ϕ obtained from Eq. 8.45 is the complex amplitude of the velocity potential. The time-dependent velocity potential is given by

$$\mathrm{Re}\left(\phi e^{i\omega t}\right) = \mathrm{Re}\left[(\phi_\mathrm{R} + i\phi_\mathrm{I})(\cos\omega t + i\sin\omega t)\right] = \phi_\mathrm{R}\cos\omega t - \phi_\mathrm{I}\sin\omega t, \tag{8.57}$$

where ϕ_R and ϕ_I are the real and imaginary parts of the complex amplitude. It is this function which is displayed in computer animations of the time-dependent response of the velocity potential.

8.7 Mechanical Analogy for the Free Surface Problem

Consider the single DOF spring-mass-dashpot system shown in Fig. 76. The application of Newton's second law of motion (F=ma) to this system yields the differential equation of motion

$$m\ddot{u} + c\dot{u} + ku = f(t), \tag{8.58}$$

where m is mass, c is the viscous dashpot constant, k is the spring stiffness, u is the displacement from the equilibrium, f is the applied force, and dots denote differentiation with respect to the time t. For a sinusoidal force,

$$f(t) = f_0 e^{i\omega t}, \tag{8.59}$$

where ω is the excitation frequency, and f_0 is the complex amplitude of the force. The displacement solution is also sinusoidal:

$$u(t) = u_0 e^{i\omega t}, \tag{8.60}$$

where u_0 is the complex amplitude of the displacement response. If we substitute the last two equations into the differential equation, we obtain

$$-\omega^2 m u_0 e^{i\omega t} + i\omega c u_0 e^{i\omega t} + k u_0 e^{i\omega t} = f_0 e^{i\omega t} \tag{8.61}$$

or

$$(-\omega^2 m + i\omega c + k)u_0 = f_0. \tag{8.62}$$

We make two observations from this last equation:

1. The inertia force is proportional to ω^2 and 180° out of phase with respect to the elastic force.

2. The viscous damping force is proportional to ω and leads the elastic force by 90°.

Thus, in the free surface problem, we could interpret the free surface matrix $\mathbf{H}_{\mathrm{FS}}$ as an inertial effect (in a mechanical analogy) with a "surface mass matrix" $\mathbf{M}$ given by

$$\mathbf{M} = \frac{1}{-\omega^2}\mathbf{H}_{\mathrm{FS}} = \frac{L}{6g}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \tag{8.63}$$

where the diagonal "masses" are positive. Similarly, we could interpret the radiation boundary matrix $\mathbf{H}_{\mathrm{RB}}$ as a "damping" effect with the "boundary damping matrix" $\mathbf{B}$ given by

$$\mathbf{B} = \frac{1}{i\omega}\mathbf{H}_{\mathrm{RB}} = \frac{\alpha L}{6\omega}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \tag{8.64}$$

where the diagonal dampers are positive. This "damping" matrix is frequency-dependent.

The free surface problem is a degenerate equivalent to the mechanical problem, since the mass $\mathbf{M}$ occurs only on the free surface rather than at every point in the domain. In fact, the ideal fluid, for which $\nabla^2\phi = 0$, behaves like a degenerate mechanical system, because the ideal fluid possesses the counterpart to the elastic forces but not the inertial forces. This degeneracy is a consequence of the incompressibility of the ideal fluid. A compressible fluid (such as occurs in acoustics) has the analogous mass effects everywhere.

Bibliography

[1] K.J. Bathe. *Finite Element Procedures*. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1996.

[2] J.W. Brown and R.V. Churchill. *Fourier Series and Boundary Value Problems*. McGraw-Hill, Inc., New York, seventh edition, 2006.

[3] G.R. Buchanan. *Schaum's Outline of Theory and Problems of Finite Element Analysis*. Schaum's Outline Series. McGraw-Hill, Inc., New York, 1995.

[4] D.S. Burnett. *Finite Element Analysis: From Concepts to Applications*. Addison-Wesley Publishing Company, Inc., Reading, Mass., 1987.

[5] R.W. Clough and J. Penzien. *Dynamics of Structures*. McGraw-Hill, Inc., New York, second edition, 1993.

[6] R.D. Cook, D.S. Malkus, M.E. Plesha, and R.J. Witt. *Concepts and Applications of Finite Element Analysis*. John Wiley and Sons, Inc., New York, fourth edition, 2001.

[7] K.H. Huebner, D.L. Dewhirst, D.E. Smith, and T.G. Byrom. *The Finite Element Method for Engineers*. John Wiley and Sons, Inc., New York, fourth edition, 2001.

[8] J.H. Mathews. *Numerical Methods for Mathematics, Science, and Engineering*. Prentice-Hall, Inc., Englewood Cliffs, NJ, second edition, 1992.

[9] K.W. Morton and D.F. Mayers. *Numerical Solution of Partial Differential Equations*. Cambridge University Press, Cambridge, 1994.

[10] J.S. Przemieniecki. *Theory of Matrix Structural Analysis*. McGraw-Hill, Inc., New York, 1968 (also, Dover, 1985).

[11] J.N. Reddy. *An Introduction to the Finite Element Method*. McGraw-Hill, Inc., New York, third edition, 2006.

[12] F. Scheid. *Schaum's Outline of Theory and Problems of Numerical Analysis*. Schaum's Outline Series. McGraw-Hill, Inc., New York, second edition, 1989.

[13] G.D. Smith. *Numerical Solution of Partial Differential Equations: Finite Difference Methods*. Oxford University Press, Oxford, England, third edition, 1985.

[14] I.M. Smith and D.V. Griffiths. *Programming the Finite Element Method*. John Wiley and Sons, Inc., New York, fourth edition, 2004.

[15] M.R. Spiegel. *Schaum's Outline of Theory and Problems of Advanced Mathematics for Engineers and Scientists*. Schaum's Outline Series. McGraw-Hill, Inc., New York, 1971.

[16] J.S. Vandergraft. *Introduction to Numerical Calculations*. Academic Press, Inc., New York, second edition, 1983.

[17] O.C. Zienkiewicz and R.L. Taylor. *The Finite Element Method for Solid and Structural Mechanics*. Elsevier Butterworth-Heinemann, Oxford, England, sixth edition, 2005.

[18] O.C. Zienkiewicz, R.L. Taylor, and P. Nithiarasu. *The Finite Element Method for Fluid Dynamics*. Elsevier Butterworth-Heinemann, Oxford, England, sixth edition, 2005.

[19] O.C. Zienkiewicz, R.L. Taylor, and J.Z. Zhu. *The Finite Element Method: Its Basis and Fundamentals*. Elsevier Butterworth-Heinemann, Oxford, England, sixth edition, 2005.

Index