Department of Civil Engineering
University of Alberta
Edmonton

Structural Engineering Report No. 149

SPREADSHEET SOLUTION OF ELASTIC PLATE BENDING PROBLEMS

by
GREGORY E. SMALL
SIDNEY H. SIMMONDS

July 1987

Structural Engineering Report No 149

Spreadsheet Solution of Elastic Plate Bending Problems

by

Gregory Eric Small
and
Sidney H. Simmonds

Department of Civil Engineering
University of Alberta
Edmonton, Alberta

July 1987

SER No 149 is based on the report entitled "Microcomputer spreadsheet solution of elastic plate bending problems using finite difference operators and iterative techniques" submitted by G. E. Small in partial fulfillment of the requirements for an M. Eng. degree.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 Introductory Remarks

Mainframe or minicomputers running finite element programs have simplified the analysis of complex slab systems. In recent years microcomputers have become powerful enough to handle many of these tasks and can be used by structural engineering consultants to augment or entirely supplant mainframe analysis. The major attraction of microcomputers is that they are relatively inexpensive to purchase, maintain, and operate. Unfortunately the software to carry out complex structural computations may not exist for a particular microcomputer or may be prohibitively expensive to purchase when compared to the cost of leasing existing programs on a mainframe system.

1.2 Objectives

This paper discusses the use of spreadsheet programs for the analysis of slab systems. The objective is to demonstrate a practical approach which can be employed for preliminary analysis, checking of mainframe solutions, or complete analysis of less complicated slab systems.

Other than word processing, the most popular use of microcomputers is for "spreadsheet" analysis of problems. The software is inexpensive, relatively easy to learn, and a highly versatile and productive tool. This paper will develop a technique which allows engineers to easily model elastic slab bending problems using readily available microcomputer spreadsheet programs.

1.3 Scope

A "central difference" formulation of the problems is used because the operators are well suited to spreadsheet modelling. Generation of the operators is illustrated using the finite difference method. A compatability approach is developed for the inclusion of beams in the model. Accuracy, convergence, and mesh size are all discussed. Several examples showing iterative solution of both classical and practical plate problems are shown.

1.4 Notation

The following notation has been adopted for use in this paper.

a	=	length of a rectangular plate in the "x" direction
b	=	length of a rectangular plate in the "y" direction or width of a rectangular beam
C	=	$\{1-0.63(b/d)\}b^3d/3$ = torsional stiffness of a rectangular beam
C_{mn}	=	Fourier coefficient for plate deflections
d	=	depth of a rectangular beam
D	=	$Et^3/\{12(1-\mu^2)\}$ = plate flexural stiffness per unit width
D_{mn}	=	Fourier coefficient for plate bending moments
E	=	modulus of elasticity of the beam and slab material
G	=	$E/\{2(1+\mu)\}$ = shear modulus of elasticity of the beam material
h	=	distance between node points
I	=	$bd^3/12$ = moment of inertia of a rectangular beam
m	=	bending moment or twisting moment per unit width of plate
$\underline{m}$	=	bending moment or twisting moment per width of plate h
M	=	bending moment in a beam
P	=	total load acting at a node point or joint
q	=	uniformly distributed load per unit of area on a plate
S	=	smoothing factor to assist convergence
t	=	thickness of plate
T	=	torsional moment in a beam
v	=	vertical shear per unit width of plate
$\underline{v}$	=	vertical shear per width of plate h
V	=	vertical shear in a beam
w	=	vertical deflection of the plate
W	=	uniformly distributed line load
{W}	=	the matrix of deflections of all nodes on the plate
x,y,z	=	rectangular reference coordinates
μ	=	Poisson's ratio

2. CENTRAL DIFFERENCE OPERATORS

2.1 Objectives

The primary purpose of this section is to provide some understanding of the generation of the central difference operators used in this paper. Central difference operators have been derived by other authors[1 2 3 4] and the reader should refer to those publications for a complete development.

2.2 Plate operators

The development of the basic equations relating plate deflections to external and internal forces is summarized as follows:

2.2.1 Governing Equations

The governing equations for elastic plate bending are derived in the literature[6] and are listed below for convenience.

Plate equations:

The deflection, w, is related to the loading by the differential equation:

$$\frac{\partial^4 w}{\partial x^4} + 2\frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} = \frac{q}{D}$$

The bending and twisting moments are related to the deflections as follows:

$$m_x = -D\left(\frac{\partial^2 w}{\partial x^2} + \mu\frac{\partial^2 w}{\partial y^2}\right)$$

$$m_y = -D\left(\frac{\partial^2 w}{\partial y^2} + \mu\frac{\partial^2 w}{\partial x^2}\right)$$

$$m_{xy} = -D(1-\mu)\frac{\partial^2 w}{\partial x \partial y}$$

Similarly, the shears are expressed as:

$$v_x = \frac{\partial m_x}{\partial x} + \frac{\partial m_{xy}}{\partial y} \quad \text{and} \quad v_y = \frac{\partial m_y}{\partial y} + \frac{\partial m_{xy}}{\partial x}$$

and the reactions at the edge of the slab are:

$$R_x = \frac{\partial m_x}{\partial x} + 2\frac{\partial m_{xy}}{\partial y} \quad \text{and} \quad R_y = \frac{\partial m_y}{\partial y} + 2\frac{\partial m_{xy}}{\partial x}$$

Beam equations:

The equations governing the deflection, flexure, shear, and twisting of beams are as follows...

beams with their longitudinal axis parallel to the x axis

$$\frac{\partial^4 w}{\partial x^4} = \frac{q_b}{EI}$$

$$M_x = -EI\frac{\partial^2 w}{\partial x^2}$$

$$V_x = -EI\frac{\partial^3 w}{\partial x^3}$$

$$T_x = -CG\frac{\partial}{\partial x}\left(\frac{\partial w}{\partial y}\right)$$

beams with their longitudinal axis parallel to the y axis

$$\frac{\partial^4 w}{\partial y^4} = \frac{q_b}{EI}$$

$$M_y = -EI\frac{\partial^2 w}{\partial y^2}$$

$$V_y = -EI\frac{\partial^3 w}{\partial y^3}$$

$$T_y = -CG\frac{\partial}{\partial y}\left(\frac{\partial w}{\partial x}\right)$$

2.2.2 Coordinate system

Central difference operators solve for deflections at the nodes of an orthogonal grid. The general form of solution allows grid lines to have different spacings in each direction. To simplify the discussion of central difference techniques, this paper will only examine operators whose grid spacings are equal in both directions. If a rectangular mesh is desired, a finer square mesh can usually be substituted to get equivalent results. Employing a finer mesh will require very little extra effort in spreadsheet fomulation although the computer will have to iterate longer to achieve convergence.

The central difference equations relate forces applied at the central node (designated node "O") to the relative deflections of surrounding nodes. Following conventions used by Ang and Prescott[7], adjacent nodes will be referenced by compass coordinates as illustrated below:

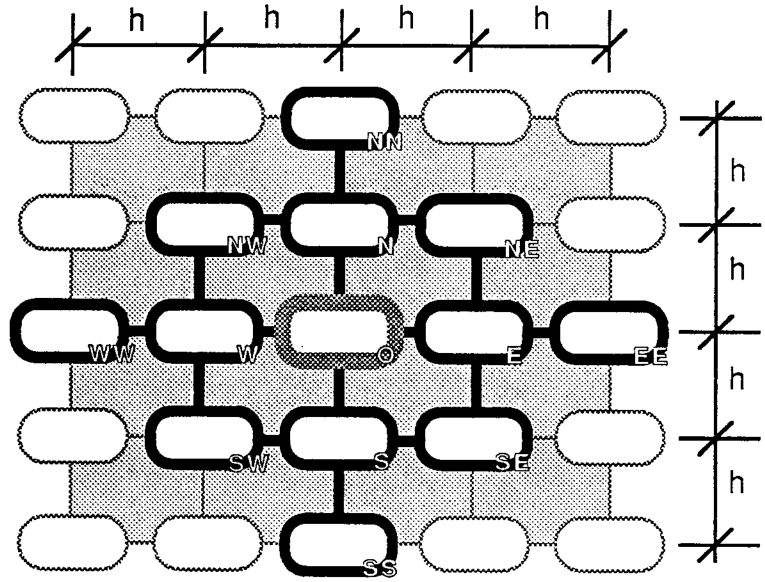


Figure 1: Compass convention for referencing nodes

where "h" is the grid spacing in both orthogonal directions.

2.2.3 Finite difference method

The finite difference technique replaces differentials with linear algebraic equations (operators) in terms of the deflections of the node points of our mesh. These can be substituted into the governing differential equations to develop operators relating applied forces to the deflections of the nodes. Derivation of the operators at point "O" can be demonstrated as follows:

w_i = the downward deflection of node "i"

$(\partial w/\partial x)_O$ = the slope of the slab at point "O" in the x direction
= the average of the slopes from nodes "W" to "O" and nodes "O" to "E"
= $[(w_O - w_W)/h + (w_E - w_O)/h]/2$
= $(w_E - w_W)/2h$

$(\partial w/\partial y)_O$ = $(w_N - w_S)/2h$

$(\partial^2 w/\partial x^2)_O$ = the curvature of the slab at point "O" in the x direction
= the rate of change of slope between "W" to "O" and "O" to "E"
$= [\,(w_E - w_O)/h - (w_O - w_W)/h\,]/h$
$= (w_E - 2w_O + w_W)/h^2$

$(\partial^2 w/\partial y^2)_O = (w_N - 2w_O + w_S)/h^2$

$(\partial^2 w/\partial x\partial y)_O = (\partial^2 w/\partial y\partial x)_O = \partial/\partial x(\partial w/\partial y)_O$
$= \partial/\partial x[\,(w_N - w_S)/2h\,]$
$= [\,\partial/\partial x(w_N) - \partial/\partial x(w_S)\,]/2h$
$= (\,w_{NE} - w_{NW} - w_{SE} + w_{SW}\,)/4h^2$

$(\partial^3 w/\partial x^3)_O = \partial/\partial x(\partial^2 w/\partial x^2)_O$
$= \partial/\partial x[\,(w_E - 2w_O + w_W)/h^2\,]$
$= [\,\partial/\partial x(w_E) - 2\partial/\partial x(w_O) + \partial/\partial x(w_W)\,]/h^2$
$= [\,(w_O - w_{EE})/2h - 2(w_W - w_E)/2h + (w_{WW} - w_O)/2h\,]/h^2$
$= (-w_{EE} + 2w_E - 2w_W + w_{WW})/2h^3$

$(\partial^3 w/\partial y^3)_O = (-w_{NN} + 2w_N - 2w_S + w_{SS})/2h^3$

similarly:

$(\partial^3 w/\partial x^2\partial y)_O = \partial/\partial y(\partial^2 w/\partial x^2)_O$
$= [\,w_{NE} + w_{NW} - w_{SW} - w_{SE} + 2(w_S - w_N)\,]/2h^3$

$(\partial^3 w/\partial x\partial y^2)_O = [\,w_{NE} - w_{NW} - w_{SW} + w_{SE} + 2(w_W - w_E)\,]/2h^3$

$(\partial^4 w/\partial x^4)_O = \partial^2/\partial x^2(\partial^2 w/\partial x^2)_O$
$= (w_{EE} - 4w_E + 6w_O - 4w_W + w_{WW})/h^4$

$(\partial^4 w/\partial y^4)_O = (w_{NN} - 4w_N + 6w_O - 4w_S + w_{SS})/h^4$

$(\partial^4 w/\partial x^2\partial y^2)_O = (\partial^4 w/\partial y^2\partial x^2) = \partial^2/\partial y^2(\partial^2 w/\partial x^2)_O$
$= [\,w_{NE} + w_{NW} + w_{SW} + w_{SE} - 2(w_N + w_S + w_E + w_W) + 4w_O\,]/h^4$

2.2.4 Finite difference solution of the general plate operator

The differential operators developed in section 2.2.3 can be substituted into the governing differential equation of plate bending given in section 2.2.1:

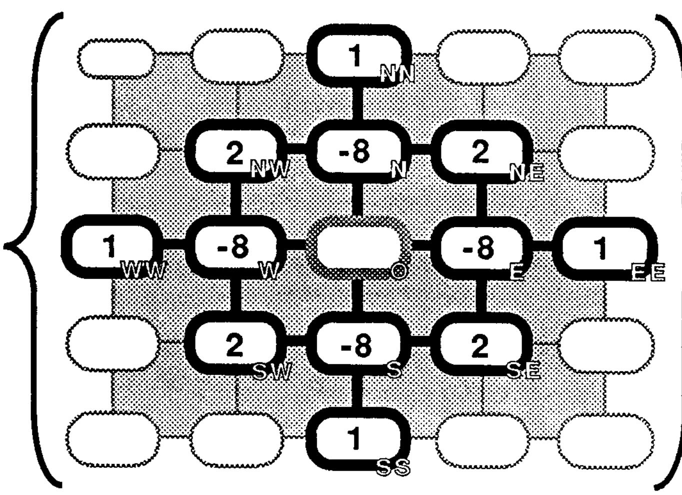
$$\frac{\partial^4 w}{\partial x^4} + 2\frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4} = \frac{q}{D}$$

to define the operator for a typical node on the plate:

$$w_O = \{ qh^4/D - [(w_{NN} + w_{SS} + w_{EE} + w_{WW}) + 2(w_{NE} + w_{NW} + w_{SW} + w_{SE}) - 8(w_N + w_S + w_E + w_W)] \}/20$$

Letting $P_o = qh^2$ = total load on a node, this can be represented graphically as follows:

$$w_o = \frac{P_o h^2}{20\,D} - \frac{1}{20}$$



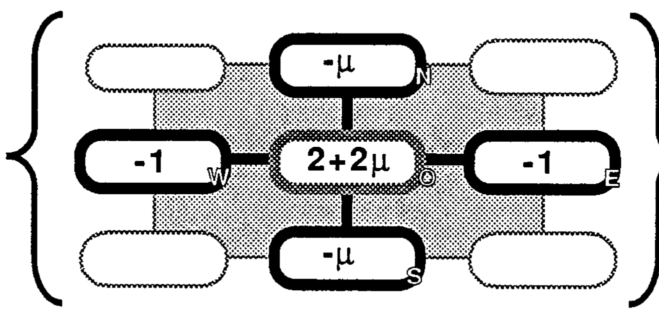
This equation calculates the deflection at node "O" as a function of the applied load "P" and deflections of the adjacent nodes. Note that this version of the general plate operator differs from the formulation used by other authors[1,2,3,4,5,7] who directly solved the simultaneous equations. This version is more suitable for spreadsheet iteration and is easily transformed algebraically to the form used by the other authors.

2.2.5 Operators for slab forces

Since the main goal of any structural analysis technique is to calculate the internal forces and boundary reactions of the system, then we also need a set of operators to generate the internal forces once the node deflections have been solved. The differential operators given in section 2.2.3 can be substituted into the differential equations for slab moments, shears, and edge reactions given in section 2.2.1. to derive the following force operators:

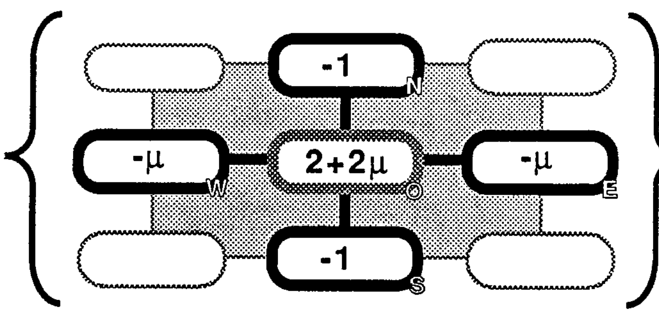
slab moments

$$m_{ox} = \frac{D}{h^2}$$



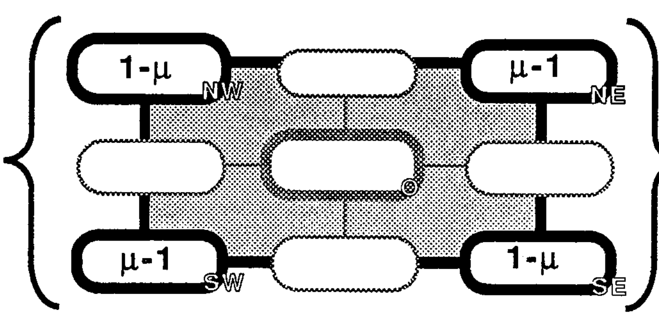
{W}

$$m_{oy} = \frac{D}{h^2}$$



{W}

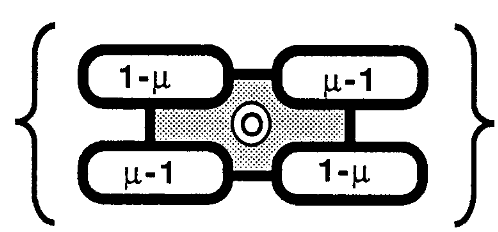
$$m_{oxy} = \frac{D}{4h^2}$$



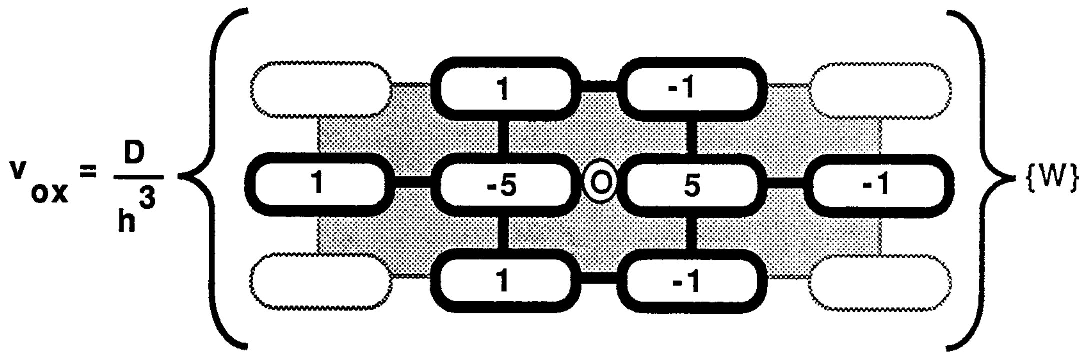
{W}

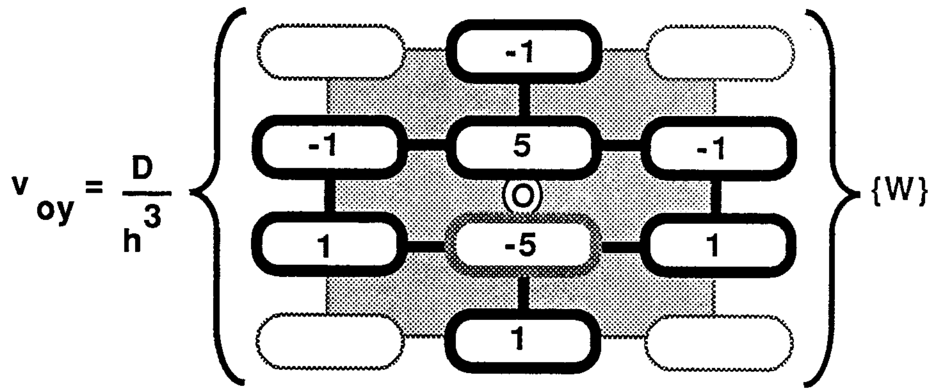
which can be "reduced" to

$$m_{oxy} = \frac{D}{h^2}$$



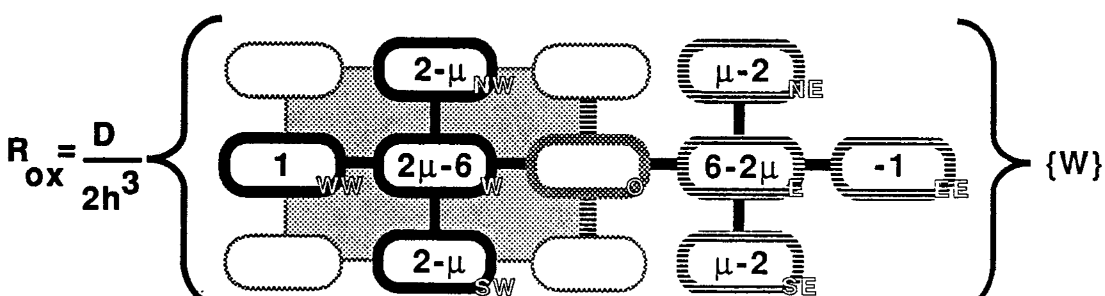
which calculates m_{xy} at point "O", centered between four nodes slab shears



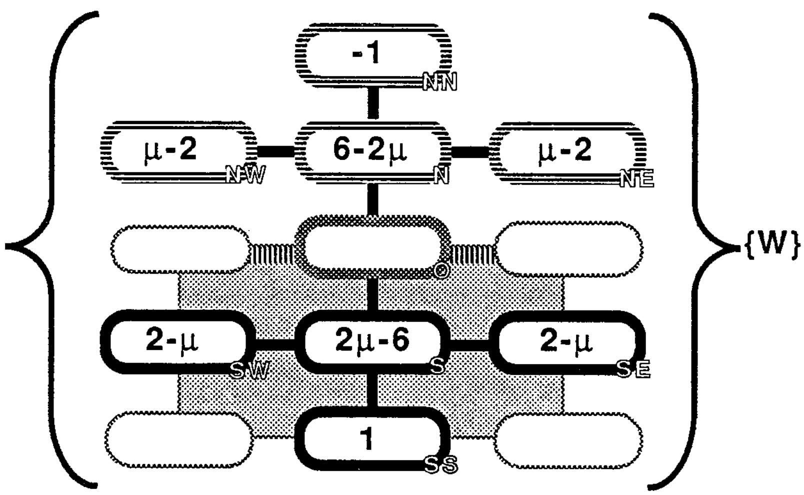


NOTE: Similar to the operator for m_{xy}, the above operators for shear produces a value at point "O" which is midway between nodes.

slab edge reactions



$$R_{oy} = \frac{D}{2h^3}$$



Notice that the operators for the edge reactions require deflections at four nodes which are not physically on the slab. These "fictitious" points are discussed in section 2.3.

2.3 Operators for plate boundary conditions

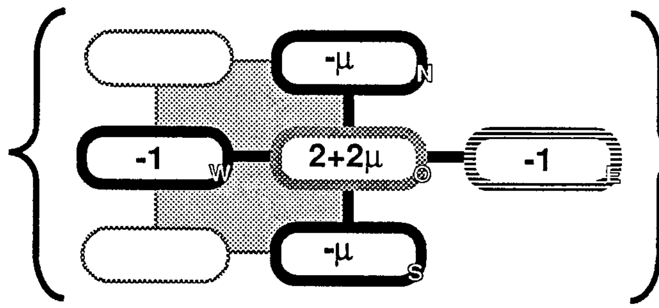
The plate operator derived in section 2.2.4 cannot, by itself, solve for the deflection of a node on or near an edge because it will "overhang" the plate boundary, seeking deflections for nodes which are not on the plate. These "fictitious" deflections describe the shape the plate would have to assume if it extended past the boundary. The moments and shears required by the real boundary conditions must be realized as internal forces in the imaginary plate extension. Operators for various boundary conditions are therefore required to solve for these "fictitious" deflections.

Note that this section will only discuss operators along the east edge of the plate. Operators for the north, south and west edges are easily obtained by rotation.

2 3.1 Boundary operator for edge moments

The operator for bending moments in the east-west direction was derived in section 2.2.5. If point "O" lies on the east boundary of the plate then "E" is a "fictitious" point:

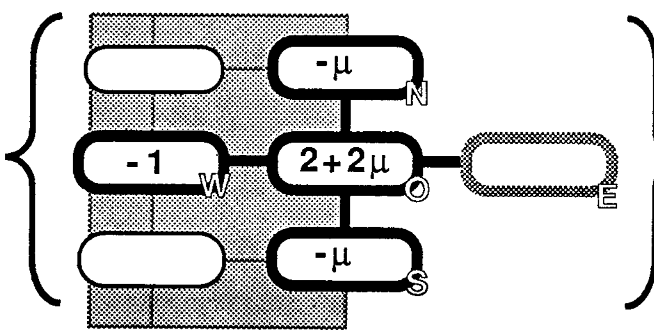
$$m_{ox} = \frac{D}{h^2}$$



$\{W\}$

If we rewrite the above equation in terms of the deflection at point "E" we create an operator which calculates the fictitious deflection in terms of the applied edge moment:

$$w_E =$$

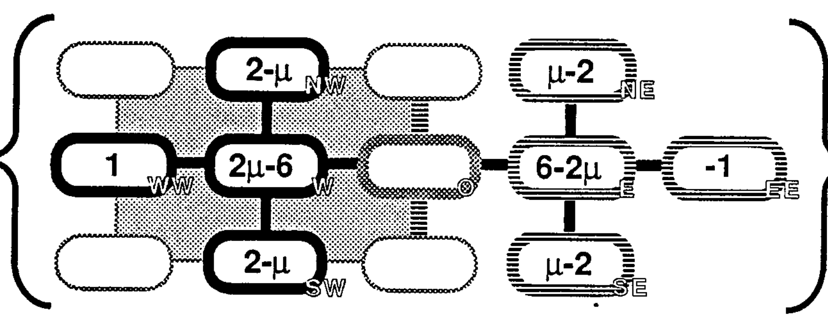


$$\{W\} - \frac{m_{ox} h^2}{D}$$

2.3.2 Boundary operator for edge reactions

The operator for edge reactions along the east edge was derived in section 2.2.5:

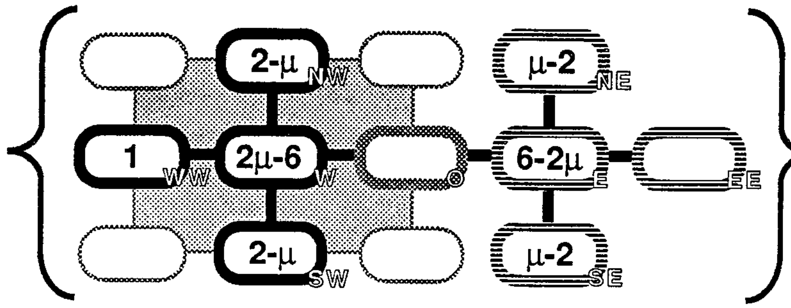
$$R_{ox} = \frac{D}{2h^3}$$



$\{W\}$

An expression for the fictitious deflection at "EE" can be written in terms of the edge reaction:

$W_{EE} =$



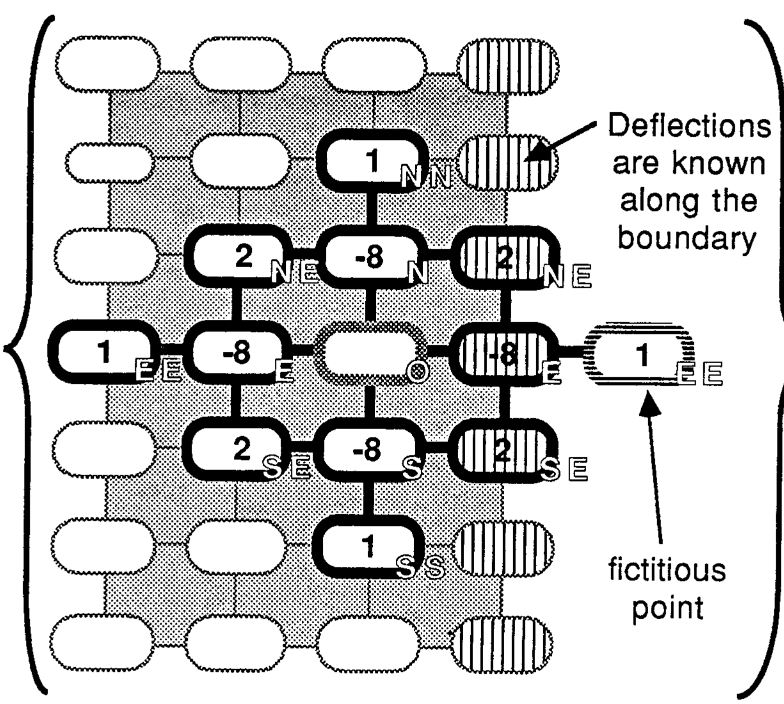
$\{W\} - 2\,\dfrac{R_{ox}h^3}{D}$

2.3.3 Boundary operators for known edge deflections

Many practical design problems involve a plate whose edges are constrained against deflection. Analysis of settlements, on the other hand, may require non zero deflections to be specified. Finally, in checking results from a finite element solution it may be useful to analyze a portion of a slab by specifying the known deflections as boundary constraints around the portion being analyzed. In all these instances the general plate operator from section 2.2.4 will be not needed at the boundary since the deflections are known explicitly.

Adjacent to any boundary with defined deflections, the general plate operator may be used but will require a fictitious deflection which is beyond the boundary:

$w_o = \dfrac{P_o h^2}{20\,D} - \dfrac{1}{20}$

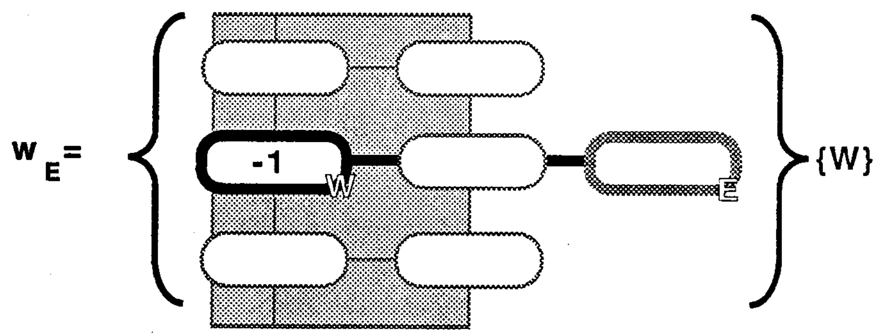


$\{W\}$

Fictitious deflection "E" is calculated using the edge moment operator from

2.3.3.1 Boundary operator for a simply supported edge

A simply supported edge specifies zero moments and deflections along the edge. Therefore, the deflection of fictitious point "E" can be calculated by using the operator from section 2.3.1 with $m_{OX} = 0$ and eliminating nodes "N", "O", and "S" whose deflections are all zero:



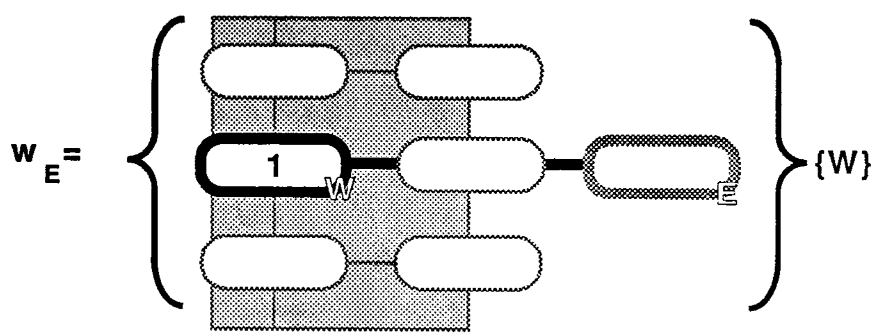
This is simply a statement of anti-symmetric deflections across the boundary.

2.3.3.2 Boundary operator for a fixed edge

A fixed edge specifies zero deflections and rotations along the edge. The differential operator for rotation (slope) is given in Section 2.2.3 as:

$$(\partial w/\partial x)_O = (w_E - w_W)/2h$$

For no rotation, $\partial w/\partial x = 0$ so $w_E = w_W$:



This is simply a statement of symmetric deflections across the boundary.

2.3.4 Customized plate operators for boundary conditions

Previous Authors[1,2,3,4,5,7,8] have found it advantageous to create custom plate operators for various boundary conditions. For some boundary conditions it is very easy to substitute the above boundary operators into the basic plate equation to eliminate the fictitious deflections. The main advantage of this approach is that it reduces the number of deflections to be solved. The main disadvantage is that a special operator needs to be developed for each boundary condition.

When it becomes necessary to incorporate the effects of beams and columns into the slab operator equation, the situation becomes much more complicated. Newmark's[8] Plate Analogue method offers an alternative approach which bypasses the differential equations by using an articulated physical model. Operator equations are easily derived using equilibrium. Interested readers should refer to the work of Prescott, Ang, and Seiss[4].

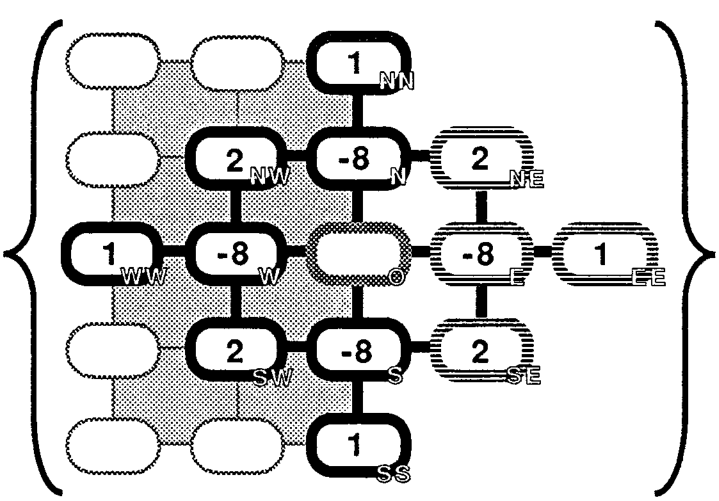
This paper will demonstrate a compatibility technique which eliminates the need for dozens of customized plate operators to handle the many combinations of beams, columns and other boundary constraints. It's possible to model any structure using only the plate operator for an interior node, combined with the operators for fictitious points derived in the previous sections. Beams, columns, etc. can be superimposed on this basic structure by calculating and applying their reactive forces.

For the solution given in this paper, it is convenient to eliminate some of the fictitious points. Two custom operators for nodes along a free edge and at a free corner are useful where an edge beam will be added to the structure.

2.3.4.1 Custom plate operator for a free edge

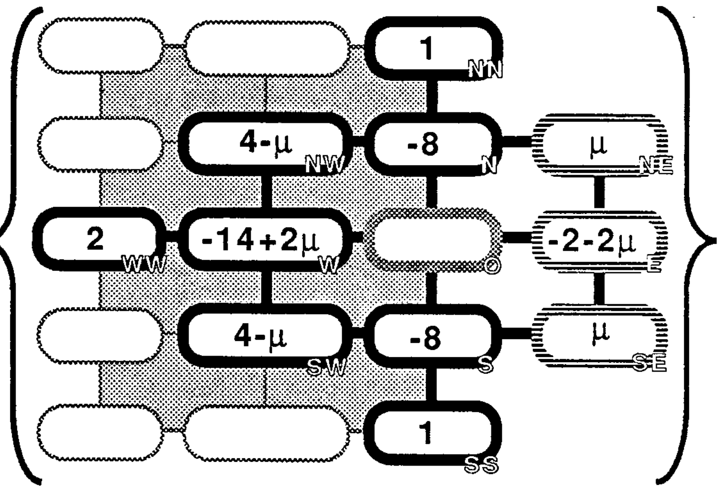
Retrieving our plate operator derived in section 2.2.4 and placing it at a node on the east edge of the plate:

$$w_o = \frac{qh^4}{20\,D} - \frac{1}{20}\left\{ \quad \right\}$$



Fictitious point "EE" can be eliminated by substitution of the edge reaction operator derived in section 2.3.2, noting that the edge reaction, "R_{OX}" is equal to zero along a free edge:

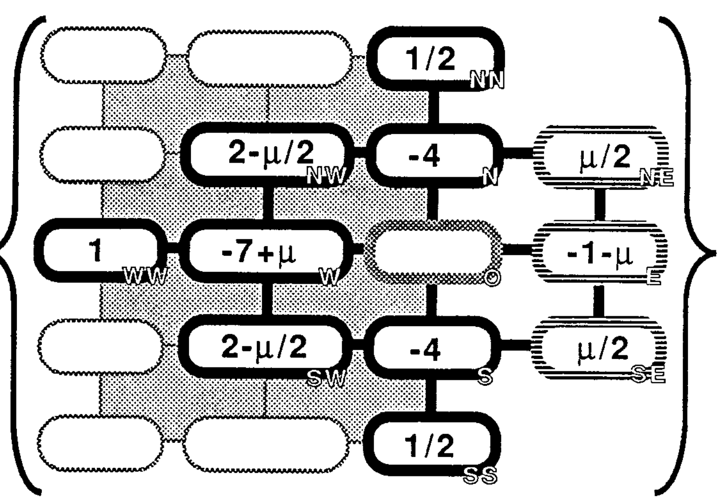
$$w_o = \frac{qh^4}{20\ D} \cdot \frac{1}{20}$$



The moment operator from section 2.3.1 could easily be used to eliminate fictitious points "NE", "E", and "SE". However, these points are required to incorporate torsional moments from an edge beam or for specifying other rotational boundary constraints. The addition of torsional edge beams to the free edge slab (section 6.4.2) uses these deflections to equate the slab edge moments to the beam torsional reactions.

As was done in section 2.2.4, it is convenient to express the operator in terms of the total load applied at the node. Since a node on the boundary has slab only on one side, the load "P" applied at the node must equal $qh^2/2$. Substituting 2P for qh^2 above gives the following operator for a free edge.

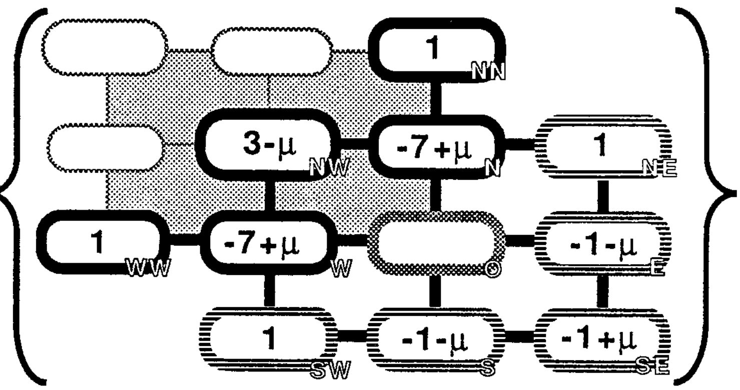
$$w_o = \frac{Ph^2}{10\ D} \cdot \frac{1}{10}$$



2.3.4.2 Custom plate operator for a free corner

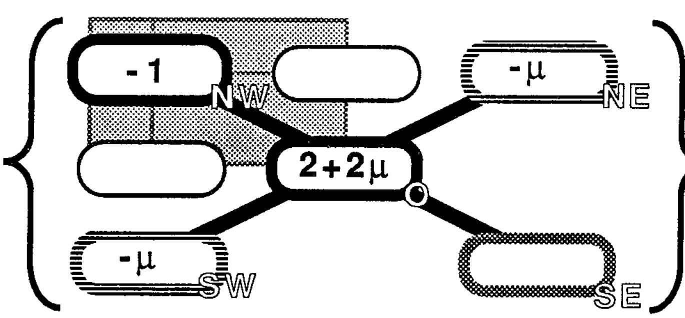
The zero edge reaction operator from section 2.3.2 can be further substituted into the free edge plate operator just developed in 2.3.4.1 to create the operator for a free corner:

$$w_o = \frac{Ph^2}{5D} - \frac{1}{10}$$



As above, deflections of nodes "NE", "E", "SE", "S", and "SW" will come from the rotational boundary constraints. Note that "SE" will require an operator which looks at slopes or moments in a diagonal direction. If the moment at "O" in the NW-SE direction is equal to $m_{\underline{OX}}$, the operator to solve for w_{SE} is easily developed by rotating the edge moment operator from section 2.3.1 by 45°:

$$w_{SE} =$$



$$\{W\} - \frac{2m_{\underline{OX}}h^2}{D}$$

Note that the node spacing becomes $h\sqrt{2}$ in the diagonal direction so that the numerator of the $m_{\underline{OX}}$ term is doubled.

$m_{\underline{OX}}$ is calculated by translation of axes for a 45° rotation:

$$m_{\underline{OX}} = (m_{OX} + m_{OY})/2 + m_{OXY}$$

2.4 Operators for beams

Many concrete designs utilize beams cast monolithically with the slab. The spreadsheet/iterative approach discussed in this paper uses compatibility of deflections to calculate the reaction forces which the beams apply to the slab.

All operators derived in this section will be for beams aligned in the east-west direction. Operators for beams aligned in other directions are easily obtained by rotation.

2.4.1 Flexural beam operators

This section develops operators which relate beam deflections to the internal bending moments and the loads applied at the node points.

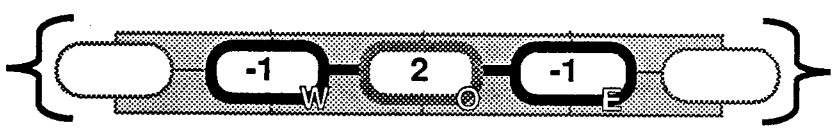
2.4.1.1 Beam bending moment operator

The differential operator for $\partial^2 w/\partial x^2$ from section 2.2.3 can easily be substituted into the differential equation of beam bending from section 2.2.1:

$$M_X = -EI\frac{\partial^4 w}{\partial x^4}$$

to define the beam bending moment operator:

$$M_{ox} = \frac{EI}{h^2}$$



$\{W\}$

2.4.1.2 Beam force/deflection operator

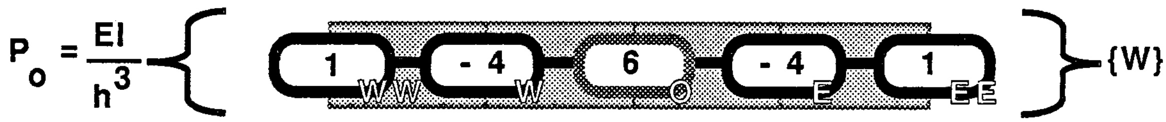
The differential operator for $\partial^4 w/\partial x^4$ from section 2.2.3 can be substituted into the governing differential equation of beam bending given in section 2.2.1:

$$\frac{\partial^4 w}{\partial x^4} = \frac{q_b}{EI}$$

to define the operator relating beam deflections to the applied load:

$$q_b h^4/EI = (w_{EE} - 4w_E + 6w_0 - 4w_W + w_{WW})$$

Letting $P_o = q_b h$ = total load on the node "O", this can be represented graphically as follows:



Conceptually, if the deflections of nodes "WW", "W", "E", and "EE" are known, the above operator defines the relationship between the force "P_o" and the deflection of node "O". As written, the operator solves for the force "P_o" required to hold node "O" at the deflection "w_o". This can be illustrated as follows:

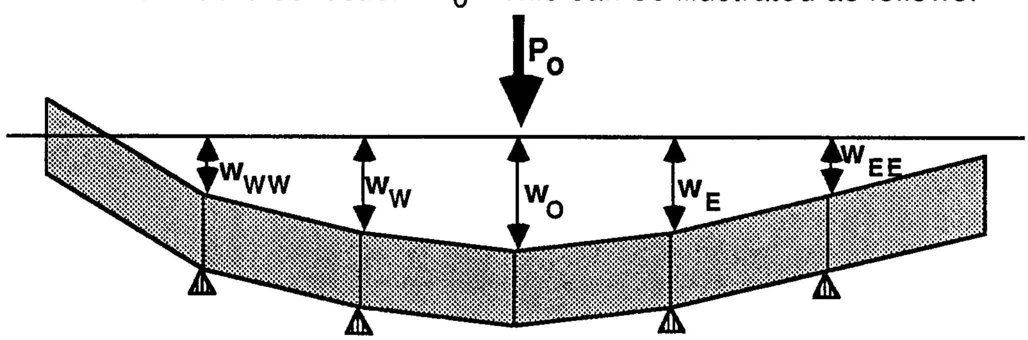


Figure 2: Deflection of node "O" under applied load P_o

2.4.2 Beam torsion operator

The differential operators developed in section 2.2.3 can be substituted into the governing differential equation of beam torsion given in section 2.2.1 :

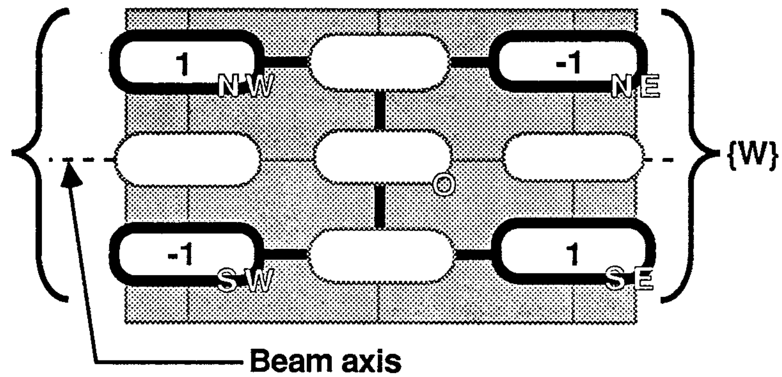
$$T_x = -CG \frac{\partial}{\partial x}\left(\frac{\partial w}{\partial y}\right)$$

to define the operator for torsion:

$$T_x = CG(w_{NW} - w_{NE} + w_{SE} - w_{SW})/4h^2$$

For a torsion beam spanning in an east-west direction, the beam operator defining torsion at point "O" would be represented as follows:

$$T_O = \frac{CG}{4h^2}$$



It would be useful to define a relationship between an applied moment "M_{OY}" at node "O" and the torsional deflections of the beam. This is analogous to the relationship between the applied load "P_O" and the deflection which was developed in the last section.

Given the deflections at the "NW", "N", "NE", "SE", "S", and "SW" nodes, the torsional equilibrium at node "O" is as follows:

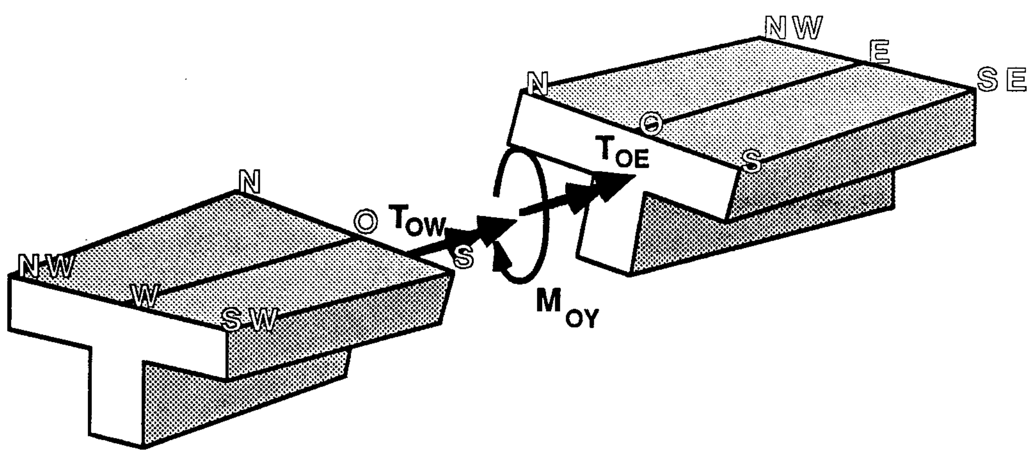
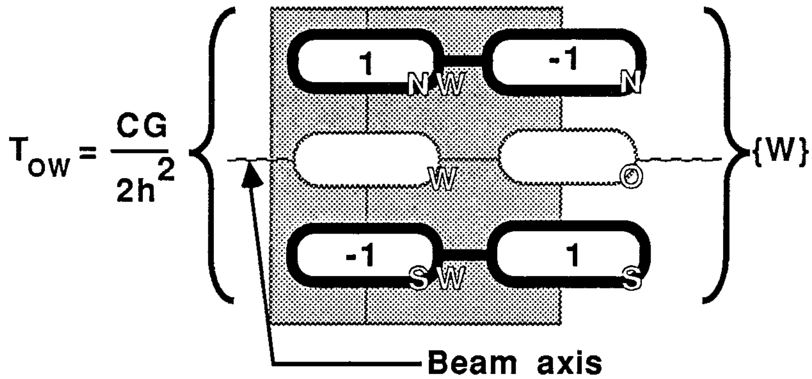
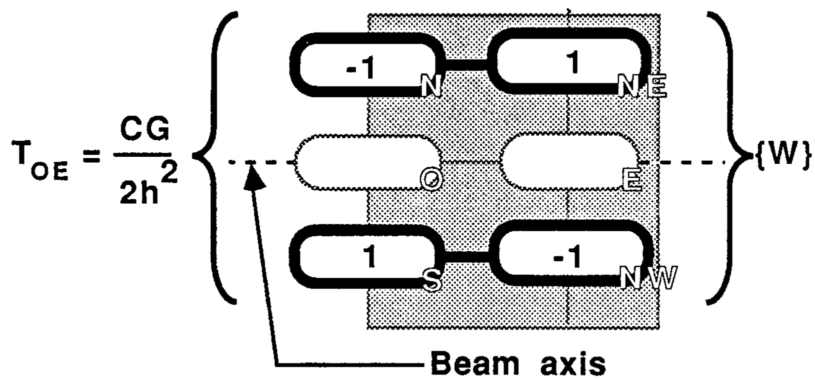


Figure 3: Torsional equilibrium at node "O"

where: $M_{OY} = T_{OW} + T_{OE}$

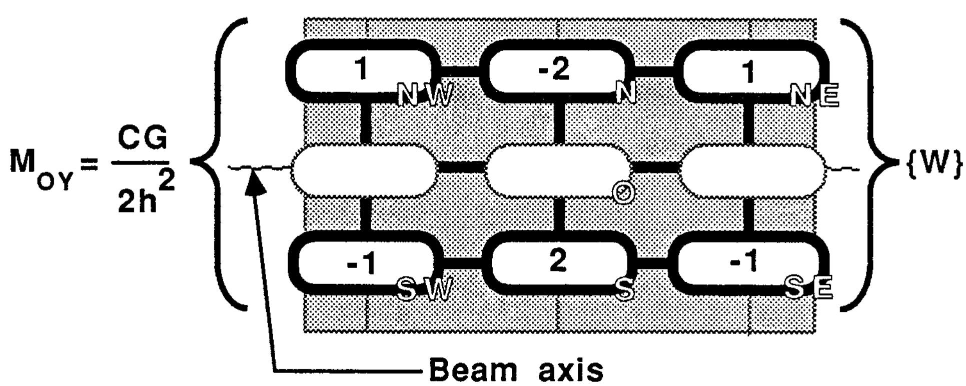
As was done in section 2.2.5 for m_{OXY}, the operator for T_O can be "shrunk" to calculate T_{OW} and T_{OE} on either side of node "O" :





Note the reversal of signs in T_{OE} due to it's left hand sign convention.

These can be summed to create the operator for the total moment, "M_{OY}", which must be applied to hold the beam in the torsionally deflected shape defined by the array {W} of node deflections adjacent to node "O" :



2.4.2.1 Operator for interior beam torsion

When a beam supports a slab, the above operator for torsion will calculate the moment which must be applied to hold the beam in it's twisted position. Since our basic plate operator developed in section 2.4.1 only relates node forces to deflections it would be useful to define M_{OY} as a pair of point loads applied at nodes "N" and "S" which straddle point "O":

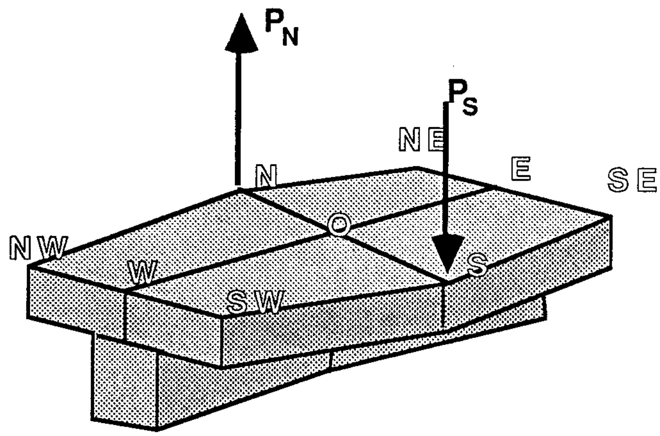
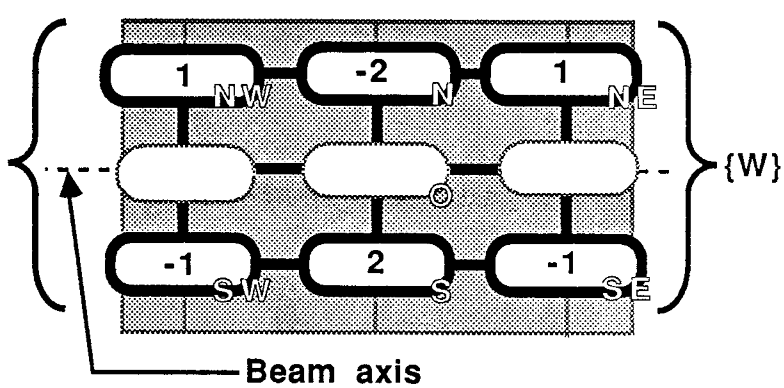


Figure 4: Force couple straddling node "O"

Noting that $P_N = P_S$ for vertical equilibrium, then $M_{OY} = 2P_Nh = 2P_Sh$, and the resulting plate operator to solve for the applied forces would be :

$$P_N = P_S = \frac{CG}{4h^3}$$



2.4.2.1 Operator for edge beam torsion

When a beam supports the outer edge of a slab, the torsional moment required to hold the beam in its torsionally deflected shape must be in equilibrium with the edge moment of the slab:

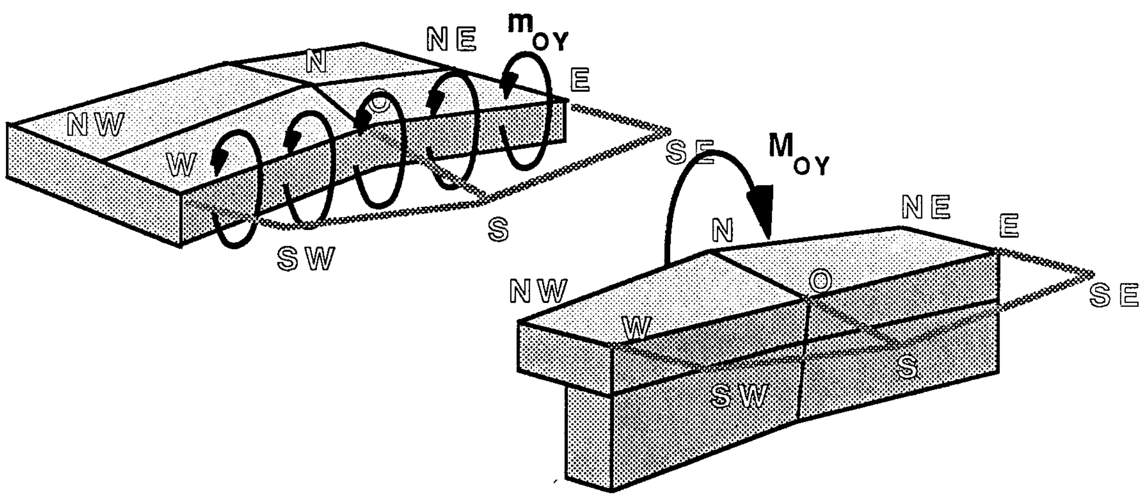
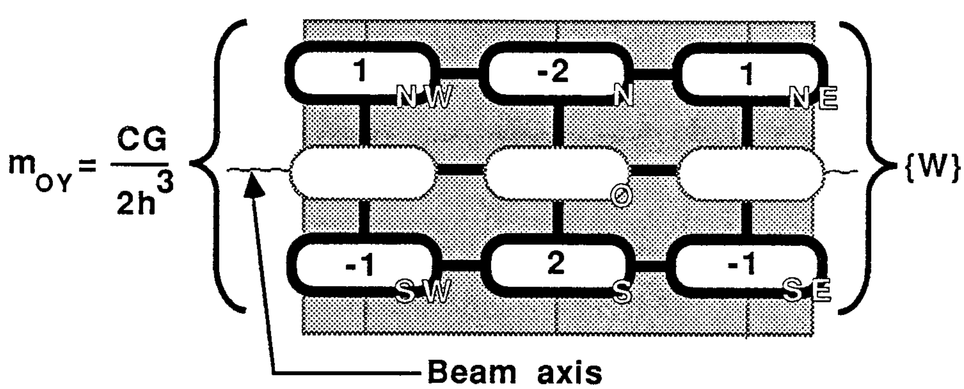


Figure 5: Moment equilibrium between the slab and edge beam

Section 2.3.1 developed an operator to calculate the deflection of the fictitious points immediately beyond the edge of the slab based on moment equilibrium. This operator will enable us to connect a torsional beam to the free edge of a slab.

Since this operator uses distributed moments, the beam torsion equation should be rewritten in terms of distributed moments as follows:



3. SOLUTIONS USING SPREADSHEET PROGRAMS

3.1 Objectives

This chapter is intended to give a basic understanding of how spreadsheet programs function, their advantages, and what features are necessary and useful when applying them to finite difference solutions. You may wish to skip section 3.2 if you are already familiar with using spreadsheet programs.

3.2 General description of spreadsheet programs

Spreadsheet programs are presented on the computer video display as a two dimensional array of boxes called **cells**. Usually there are many more cells in the spreadsheet than can be shown at one time. Using **cursor keys** (or a pointing device such as a **mouse**) the user is able to **scroll** the displayed region of the spreadsheet to view any particular cell or cells.

Just as elements of a matrix are addressed by their column and row locations, each column and row of a spreadsheet is referred to by a letter or number. A **reference** to a particular cell is made by specifying the appropriate column and row **address**. For example, "F4" or "R4C6" would each refer to cell which is at the intersection of the sixth (Fth) column to the right and the fourth row down. This paper will use the Letter–Number ("F4") reference system for specifying the column and row addresses of cells.

Each cell is like a pigeon hole which may contain a **number**, a **formula** or some **text**. When a cell contains a formula it performs a calculation and will display the resulting number. A formula can make use of numbers in adjacent cells. For example, a formula in cell "A8" could be "=A5+A6+A7" which would calculate and display the sum of three numbers whose values are found in cells "A5", "A6", and "A7".

A powerful feature of spreadsheet formulas is that they can employ **fixed addressing** or **relative addressing** to reference a cell's location. A fixed address, designated in this paper by a "$" sign, points to an absolute location on the spreadsheet (ie. the J^{th} column and 4^{th} row is shown as "$J$4"). A relative address locates a particular cell by noting its distance from the cell containing the formula (ie 2 rows up and 3 columns right is designated "J4" when referenced from cell G6). It's even possible to mix the addressing schemes within a cell reference ($J4 or J$4). A formula can be **copied** to another cell to perform the same calculation but on different numbers. For instance, the formula in cell "A8" given in the last paragraph could be copied to cell "C10". If relative addressing had been used "C10" would contain the formula "=C7+C8+C9" and would display that sum. If fixed addressing had been employed then "C10" would contain the formula "=$A$5+$A$6+$A$7" and would calculate the same number as displayed in cell "A8". Relative and fixed addressing can be intermixed within a formula so that constants such as Young's Modulus stay referenced to an absolute location on the spreadsheet but the rest of the formula "moves" as it is copied.

3.3 Spreadsheets applied to solution of central difference equations

There are several characteristics of spreadsheets which make them ideal for formulating central difference solutions:

- The nodes of a central difference mesh are located on a rectangular grid which matches nicely with the grid of cells available on a spreadsheet. This permits the user to graphically visualize the mesh on the computer screen.

- Many spreadsheet programs allow the user to select cells with cursor control keys (or a pointing device such as a mouse) to enter the cell reference into a formula. To create a finite difference operator the user can "point" to the nodes of the mesh while typing in appropriate constants and mathematical functions from the keyboard. Thus the user is able to "visualize" the nodes in an operator as it is created.

- The central difference operators are often identical for many nodes in the mesh. A "relatively addressed" formula can be efficiently copied to other cells (nodes) requiring the same operator. This greatly reduces the amount of time needed to create the spreadsheet model.

3.4 Special requirements for a spreadsheet central difference solution

The finite difference operators developed in chapter 2 calculate the deflection of node "O" based on the deflections of adjacent nodes. Unfortunately, the deflection of these adjacent nodes reciprocally depend on the deflection at node "O". A direct solution is not possible since the equations are mutually dependant. The classical approach to finite difference solutions involves setting up a simultaneous equations matrix and solving for the coefficients. This approach is very cumbersome to implement on a spreadsheet and would require a sophisticated program to rework the finite difference formulae and set up the equations in matrix form. Furthermore, routines for solution of the matrix would also have to be developed by the user since few spreadsheet programs contain matrix functions.

A better approach is to let the computer iteratively calculate the system of interdependent operators until convergence is achieved. This may require thousands of iterations, tying up the computer for tens of minutes. Nonetheless, this technique is easily understood and is simple to implement.

The techniques described in this paper require the use of a spreadsheet program which can iterate to solve dependant formulae.

3.5 Selection of a spreadsheet program

Some of the more popular spreadsheet programs which are suitable for the formulation described in this paper include Multiplan®, Lotus 1-2-3® and Excel®. There are very few microcomputers that will not run one of these programs.

Microsoft Excel® running on the MacIntosh® microcomputer was chosen for use in preparing this paper because it has the following features:

- a direct interface with MacIntosh word processing programs so that a drawing of the screen display can be "pasted" into the word processing documents to illustrate the text.

- the display of individual cells can be modified (highlighted, italicized, and framed) to enhance the "visual" aspect of spreadsheet formulation.

- powerful charting functions to graphically illustrate the results of calculations are included with the program.

- complete control of iteration parameters such as the desired accuracy of the solution and the maximum number of iterations to be tried. (Note to Lotus 1-2-3® users: You can only specify a maximum of 50 iterations. Some solutions discussed in this paper will need up to 10,000 iterations so you will probably want to develop a macro to repeatedly restart the iterations.)

- the user can create " command key macros " to automate tasks such as entering central difference operators into cells. A macro which set all the deflections of the plate to zero before commencing iterations was particularly useful to the Author since it provided a uniform starting point when comparing the number of iterations required to get a solution. More importantly, the zeroing macro permitted easy recovery if the solution started to diverge due to an error in the spreadsheet equations.

- custom functions can be written to supplement the mathematical operations which are included with the spreadsheet program. The Author created "function" macros to solve systems of simultaneous equations and Fourier summations of plate deflections.

All solutions illustrated in this paper were performed on a MacIntosh® 512k enhanced microcomputer using the Excel® spreadsheet program. Time required to achieve the solutions will vary for other microcomputer systems or other configurations of the MacIntosh®. Microcomputers such as the IBM PC–AT® or MacIntosh II®, which use more powerful processors and math co-processors, will solve the same spreadsheets in a fraction of the times quoted in this paper. Some of the spreadsheet solutions developed in chapters 4, 5, and 6 were translated by Excel® into Lotus 1–2–3® format and were successfully transferred to and solved on an IBM PC®.

4. SOLUTION OF A SIMPLY SUPPORTED PLATE

4.1 Objectives

This chapter will illustrate how to use a spreadsheet program to calculate the deflections and stresses in a simply supported square plate under uniform loading. It will explore the convergence of the iterative finite difference formulation, comparing the accuracy to a classical Navier solution. Mesh fineness and taking advantage of symmetry will also be discussed.

4.2 A square, simply supported plate

Starting off with a "simple" example, the following square slab on knife edge supports will be analyzed for deflections and moments:

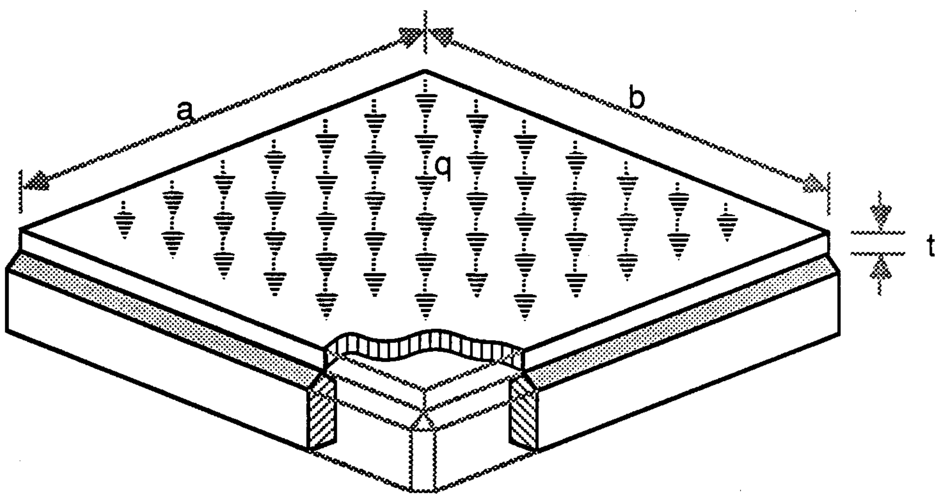


Figure 6: Simply supported square plate under uniform load

where:	slab thickness	= t	= 0.30 m
	slab width & breadth	= a = b	= 6.00 m
	uniformly distributed load	= q	= 15.0 kPa

The slab is constructed out of concrete with the following properties:

Young's modulus	= E	= 27 GPa
Poisson's ratio	= μ	= 0.15

And the slab stiffness is therefore calculated as:

flexural stiffness per unit width	= D	= 62.15 MN-m

4.3 Navier's Solution

An accepted plate bending solution is needed to assess the accuracy of the spreadsheet formulation. A Fourier series for a simply supported plate under uniform loading was developed by Navier[9] and is given below:

$$w = \sum_m \sum_n \left\{ C_{mn} \, \mathrm{Sin}\left(\frac{m\pi x}{a}\right) \mathrm{Sin}\left(\frac{n\pi y}{b}\right) \right\}$$

$$m_x = \sum_m \sum_n \left\{ D_{mn} \, \mathrm{Sin}\left(\frac{m\pi x}{a}\right) \mathrm{Sin}\left(\frac{n\pi y}{b}\right) \right\}$$

where:

$$C_{mn} = \left\{ \frac{16\,q}{\pi^6 D} \right\} \left(\frac{1}{mn\left[\left(\frac{m}{a}\right)^2 + \left(\frac{n}{b}\right)^2\right]^2} \right)$$

$$D_{mn} = \left\{ \frac{16\,q}{\pi^4} \right\} \left(\frac{\left(\frac{m}{a}\right)^2 + \mu\left(\frac{n}{b}\right)^2}{mn\left[\left(\frac{m}{a}\right)^2 + \left(\frac{n}{b}\right)^2\right]^2} \right)$$

and:

$m = 1,3,5,7,.......$

$n = 1,3,5,7,.......$

The above equations were programmed as spreadsheet functions of the following parameters:

w = function (q, a, b, D, x, y, terms)

mx = function (q, a, b, μ, x, y, terms)

where: q, a, b, D, and μ are defined as above

x and y define the location on the plate

"terms" defines the number of Fourier harmonics to be summed

These spreadsheet Navier functions were used to calculate the deflections and moments at 0.6 m grid intervals. The Fourier series summed all combinations of the first 5 terms (n, m = 1,3,5,7,9). The number of locations where calculations had to be carried out was reduced by taking advantage of the double symmetry. A copy of the spreadsheet is shown below:

Simply supported slab, Navier's Solution

Slab width: a =	6.00	m	Young's modulus: E =	27	GPa
Slab depth: b =	6.00	m	Poisson's ratio: μ =	0.15	
Slab thickness: t =	0.30	m	Slab Stiffness: D =	62	MN-m
Uniform load: q =	15.00	kPa	Fourrier term depth =	5	

Slab deflection (mm)

x =	0.0	0.6	1.2	1.8	2.4	3.0	3.6	4.2	4.8	5.4	6.0
y =											
0.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0.6	0.0000	**0.1360**	**0.2523**	**0.3397**	**0.3934**	**0.4115**	0.3934	0.3397	0.2523	0.1360	0.0000
1.2	0.0000	0.2523	**0.4701**	**0.6346**	**0.7360**	**0.7703**	0.7360	0.6346	0.4701	0.2523	0.0000
1.8	0.0000	0.3397	0.6346	**0.8583**	**0.9968**	**1.0436**	0.9968	0.8583	0.6346	0.3397	0.0000
2.4	0.0000	0.3934	0.7360	0.9968	**1.1585**	**1.2133**	1.1585	0.9968	0.7360	0.3934	0.0000
3.0	0.0000	0.4115	0.7703	1.0436	1.2133	**1.2707**	1.2133	1.0436	0.7703	0.4115	0.0000
3.6	0.0000	0.3934	0.7360	0.9968	1.1585	1.2133	1.1585	0.9968	0.7360	0.3934	0.0000
4.2	0.0000	0.3397	0.6346	0.8583	0.9968	1.0436	0.9968	0.8583	0.6346	0.3397	0.0000
4.8	0.0000	0.2523	0.4701	0.6346	0.7360	0.7703	0.7360	0.6346	0.4701	0.2523	0.0000
5.4	0.0000	0.1360	0.2523	0.3397	0.3934	0.4115	0.3934	0.3397	0.2523	0.1360	0.0000
6.0	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

mx (KN-m/m)

x =	0.0	0.6	1.2	1.8	2.4	3.0	3.6	4.2	4.8	5.4	6.0
y =											
0.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.6	0.00	**4.14**	**5.97**	**7.08**	**7.51**	**7.72**	7.51	7.08	5.97	4.14	0.00
1.2	0.00	**7.01**	**10.70**	**12.83**	**13.76**	**14.13**	13.76	12.83	10.70	7.01	0.00
1.8	0.00	**8.96**	**14.07**	**17.07**	**18.45**	**18.95**	18.45	17.07	14.07	8.96	0.00
2.4	0.00	**10.05**	**16.05**	**19.62**	**21.31**	**21.90**	21.31	19.62	16.05	10.05	0.00
3.0	0.00	**10.43**	**16.71**	**20.49**	**22.28**	**22.91**	22.28	20.49	16.71	10.43	0.00
3.6	0.00	10.05	16.05	19.62	21.31	21.90	21.31	19.62	16.05	10.05	0.00
4.2	0.00	8.96	14.07	17.07	18.45	18.95	18.45	17.07	14.07	8.96	0.00
4.8	0.00	7.01	10.70	12.83	13.76	14.13	13.76	12.83	10.70	7.01	0.00
5.4	0.00	4.14	5.97	7.08	7.51	7.72	7.51	7.08	5.97	4.14	0.00
6.0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Figure 7: Navier's solution of a square simply supported elastic plate

4.4 Symmetry and mesh layout considerations

In section 4.3 we summed only 25 terms of the Fourier series at each point on the 10 x 10 mesh. This involved a monumental calculation, tying up the Author's MacIntosh® for over four minutes. For all this effort, the solution only summed enough terms to give about 0.1% ($1/11^3$) accuracy. While this is well beyond the precision required for practical engineering, it is needed to calculate the relative accuracy of the spreadsheet/iterative solutions.

The finite difference formulation discussed in this paper faces a similar tradeoff between the accuracy of our solutions and our computing resources. A general principle for descretized solutions of continuum problems is that the results become more accurate as the mesh is refined. This is known to be true for finite element solutions[10] and is intuitively correct for finite difference formulations. Section 4.5.1 will explore mesh refinement versus accuracy and computing time considerations.

Careful consideration of how we define our mesh, and taking advantage of any symmetry which is available, will allow us to have a fairly refined mesh with a minimum of unknowns to be solved by the computer. For instance, our 6m by 6m slab could be modelled as a four by four mesh for analysis using finite differences. We know that the deflections along the edges of the slab are zero so we only need to solve the interior nodes. If the applied load is non uniform, or if symmetry is simply ignored, we would need to solve for nine unknown deflections as follows:

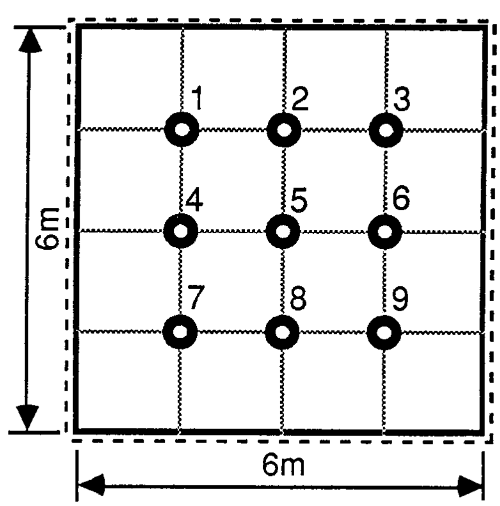


Figure 8: Nodes on a 4x4 mesh

However, if we are able to take advantage of the three-way symmetry, we can see that there are really only three deflections which need to be solved. All other nodes have identical deflections as numbered below:

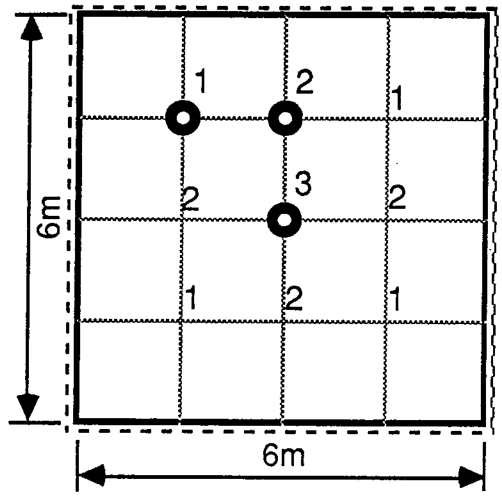


Figure 9: Three-way symmetry on a 4x4 mesh

But can we use our three unknowns more effectively? If we select a 5 x 5 mesh then we have refined our mesh by 78% (4x4 vs 3x3 deflected nodes). Notice that with double symmetry we still have only three unknowns:

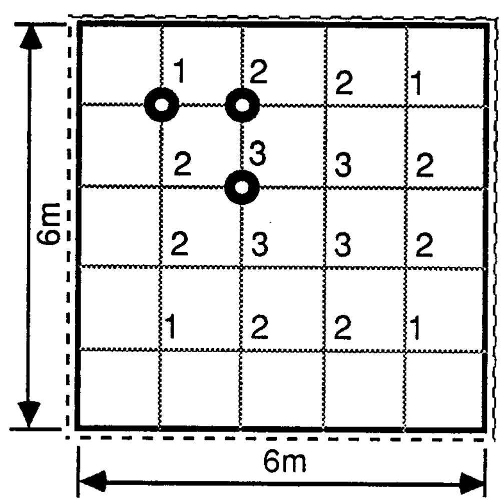


Figure 10: Three-way symmetry on a 5x5 mesh

In general, when there is symmetry, an odd sized mesh (3x3, 5x11, 7x9) will provide the most efficient use of unknowns. Note however that the midspan deflections will not be solved. It may be desirable to sacrifice the extra efficiency in order to accurately obtain the (maximum) moments and deflections which occur at midspan.

4.5 Step by step solution using a spreadsheet

The 6m x 6m slab can be modelled with finite differences on a 5x5 mesh, employing a spreadsheet program as follows:

Step 1: Open a new worksheet and enter the problem parameters near the top.

	A	B	C	D	E	F	G	H	I	J
1	Finite difference solution of square slabs using a 5x5 mesh									
2										
3	Grid spacing: h =			1.20	m	Uniform load: q =			15.00	kPa
4	Slab thickness: t =			0.30	m	Grid point load: P =			21.60	kN
5	Young's modulus: E =			27.00	GPa	Slab Stiffness: D =			62148	kN-m
6	Poisson's ratio: μ =			0.15			Ph^2/D =		0.5005	mm
7										
8										

Notice that: h = 6m÷5 = 1.20 m for the 5x5 mesh.

The following parameters were calculated by the spreadsheet when appropriate formulas were entered into cells "I4", "I5", and "I6":

$P = qh^2 = 21.60$ kN

$D = Et^3/\{12(1-\mu^2)\} = 62148$ kN-m

$Ph^2/D = 0.00050048$ m. This was multiplied by 1000 to get mm.

Step 2: Set up the zero deflections along the plate boundary to outline the region which contains the nodes of the mesh:

	A	B	C	D	E	F	G	H	I	J
1	Finite difference solution of square slabs using a 5x5 mesh									
2										
3	Grid spacing: h =			1.20	m	Uniform load: q =			15.00	kPa
4	Slab thickness: t =			0.30	m	Grid point load: P =			21.60	kN
5	Young's modulus: E =			27.00	GPa	Slab Stiffness: D =			62148	kN-m
6	Poisson's ratio: μ =			0.15				Ph^2/D =	0.5005	mm
7										
8										
9		A: DEFLECTIONS: w (mm)								
10										
11										
12			0	0	0	0	0	0		
13			0					0		
14			0					0		
15			0					0		
16			0					0		
17			0	0	0	0	0	0		
18										

Step 3: Enter the formula for the plate operator into cell D13

	B	C	D	E	F	G	H	I
9	A: DEFLECTIONS: w (mm)							
10								
11								
12		0	0	0	0	0	0	
13		0	0.0250				0	
14		0					0	
15		0					0	
16		0					0	
17		0	0	0	0	0	0	
18								

The formula entered into cell D13 was the general plate operator developed in section 2.2.4:

"=($I$6-(D11+F13+D15+B13+2*(C12+E12+E14+C14)-8*(D12+E13+D14+C13)))/20"

If the spreadsheet program allows selection of cells with a pointing device such as a mouse, formula creation is greatly simplified. Instead of having to figure out the address of a cell and typing it, you simply point to, and click on cell you want. This is much faster and gives a visual feel to the formula.

Notice that the formula which was placed in cell D13 has both relative and fixed addressing of the cells. No matter where our formula is copied, we always need to reference the value of Ph^2/D which is found in cell $I$6. The remaining cell references are free to move with the plate operator formula as it is copied to other grid points in our model.

The number "0.0250" displayed in cell D13 is the result of calculating our plate operator formula with all surrounding deflections equal to zero and is simply $Ph^2/20D$.

Step 4: Copy the plate operator formula to the other nodes where deflections must be solved:

	B	C	D	E	F	G	H	I
11								
12		0	0	0	0	0	0	
13		0	0.0250	0.0350			0	
14		0		0.0365			0	
15		0					0	
16		0					0	
17		0	0	0	0	0	0	
18								

As shown in section 3.2.2 we only need to solve for the deflections at three points on our 5x5 mesh since the rest are known by symmetry.

At this point most spreadsheet programs will warn the user that they "cannot resolve the circular references". This is because the formula in E13 depends on the value in D13 which itself uses the value in E13, and so forth. The user should select the spreadsheet options to carry out iterative calculations to solve these dependant equations.It would also be worthwhile to turn off automatic recalculation to prevent time wasting iterations after each change to the worksheet during the development of our model. If manual recalculation can be specified, iterations will not begin until requested by the user.

The values displayed in cells E13 and E14 are based on the values of the adjacent cells when the formula from D13 was copied into them. The interested reader may wish to verify that E13 received the formula before E14 .

Step 5: Enter formulas in the remaining nodes to reflect the symmetry of the deflections:

This is necessary because the plate operators need the deflections at these nodes.

	B	C	D	E	F	G	H	I
11								
12		0	0	0	0	0	0	
13		0	**0.0250**	**0.0350**	0.0350	0.0250	0	
14		0	0.0350	**0.0365**	0.0365	0.0350	0	
15		0	0.0350	0.0365	0.0365	0.0350	0	
16		0	0.0250	0.0350	0.0350	0.0250	0	
17		0	0	0	0	0	0	
18								

A typical symmetry formula is "=E14" which is found in cells F14, E15, and F15. Note that we didn't really require values for cells G15, D16, F16, and G16 since our operator formulas don't reach them. They were included for completeness of the solution but can be omitted to make the iterations slightly faster.

Step 6: Enter formulae for the fictitious points:

The plate operator formulas need deflections from three points which lie outside the plate boundaries. Section 2.3.3.1 shows that for a simply supported edge, these fictitious points are antisymmetric reflections across the plate boundary.

	B	C	D	E	F	G	H	I
9	**A: DEFLECTIONS: w (mm)**			number of iterations/10000 =				0.0000
10								
11			*-0.025*	*-0.035*				
12		0	0	0	0	0	0	
13	*-0.025*	0	**0.0250**	**0.0350**	0.0350	0.0350	0	
14		0	0.0350	**0.0365**	0.0365	0.0350	0	
15		0	0.0350	0.0365	0.0365	0.0350	0	
16		0	0.0350	0.0350	0.0350	0.0350	0	
17		0	0	0	0	0	0	
18								

The formula for the fictitious point E11 is simply "=–E13". The other two fictitious points contain the formula "=–D13".

Notice that there is an iteration counting formula in cell I9 to give additional information about the calculations. This is strictly for the purposes of this study and is not a necessary part of the solution.

Step 7: Iterate to a solution:

All that remains is to begin the iterations. But how far should the iterations continue? Excel® allows the user to specify the number of iterations and/or the desired convergence accuracy, dictated by a test which looks for a maximum change between subsequent iterations.

In order to explore the accuracy of results against the number of iterations, the computer was asked to iterate to a specified convergence. The iteration counting function kept track of the cumulative iterations.

a) iterate to 0.1 decimal convergence:

	B	C	D	E	F	G	H	I
9	**A: DEFLECTIONS: w (mm)**			number of iterations/10000 =				0.0001
10								
11			*-0.025*	*-0.035*				
12		0	0	0	0	0	0	
13	*-0.025*	0	**0.0484**	**0.0645**	0.0645	0.0484	0	
14		0	0.0645	**0.0839**	0.0839	0.0645	0	
15		0	0.0645	0.0839	0.0839	0.0645	0	
16		0	0.0484	0.0645	0.0645	0.0484	0	
17		0	0	0	0	0	0	
18								

- The calculations stopped after only one iteration because there was no change in the first decimal place in any of the numbers. Note that there were very large changes in the second decimal place so a finer convergence accuracy needs to be specified.

b) iterate to 0.01 decimal convergence:

	B	C	D	E	F	G	H	I
9	**A: DEFLECTIONS: w (mm)**			number of iterations/10000 =				0.0037
10								
11			*-0.379*	*-0.588*				
12		0	0	0	0	0	0	
13	*-0.379*	0	**0.3830**	**0.5946**	0.5946	0.3830	0	
14		0	0.5946	**0.9277**	0.9277	0.5946	0	
15		0	0.5946	0.9277	0.9277	0.5946	0	
16		0	0.3830	0.5946	0.5946	0.3830	0	
17		0	0	0	0	0	0	
18								

- After 37 iterations, taking less than 2 seconds, the results are about 25% less than the Navier solution given on page 23 for grid points (1.2 m, 1.2 m), (2.4 m, 1.2 m), and (2.4 m, 2.4 m).

c) iterate to 0.001 decimal convergence:

	B	C	D	E	F	G	H	I
9	A: DEFLECTIONS: w (mm)number of iterations/10000 =							0.0091
10								
11			*-0.464*	*-0.722*				
12		0	0	0	0	0	0	
13	*-0.464*	0	**0.4640**	**0.7230**	0.7230	0.4640	0	
14		0	0.7230	**1.1319**	1.1319	0.7230	0	
15		0	0.7230	1.1319	1.1319	0.7230	0	
16		0	0.4640	0.7230	0.7230	0.4640	0	
17		0	0	0	0	0	0	
18								

- After 3 more seconds of iteration the results are now within 2.5%.

d) iterate to 0.0001 decimal convergence:

	B	C	D	E	F	G	H	I
9	A: DEFLECTIONS: w (mm)number of iterations/10000 =							0.0144
10								
11			*-0.472*	*-0.735*				
12		0	0	0	0	0	0	
13	*-0.472*	0	**0.4718**	**0.7354**	0.7354	0.4718	0	
14		0	0.7354	**1.1517**	1.1517	0.7354	0	
15		0	0.7354	1.1517	1.1517	0.7354	0	
16		0	0.4718	0.7354	0.7354	0.4718	0	
17		0	0	0	0	0	0	
18								

- After 3 more seconds the cumulative totals are 144 iterations in 8 seconds. The values agrees with Navier's solution to within 0.6%.

e) iterate to 0.00001 convergence:

	B	C	D	E	F	G	H	I
9	A: DEFLECTIONS: w (mm)number of iterations/10000 =							0.0391
10								
11			*-0.473*	*-0.737*				
12		0	0	0	0	0	0	
13	*-0.473*	0	**0.4727**	**0.7368**	0.7368	0.4727	0	
14		0	0.7368	**1.1539**	1.1539	0.7368	0	
15		0	0.7368	1.1539	1.1539	0.7368	0	
16		0	0.4727	0.7368	0.7368	0.4727	0	
17		0	0	0	0	0	0	
18								

- Fourteen more seconds of iteration did not appreciably improve the accuracy. It is apparant that we have iterated to the limiting accuracy of the plate analogue descretization of this problem. Any further increase in accuracy will require a finer mesh.

Step 8: Solve for m_x based on the calculated deflections:

The plate operator for m_x is given in section 2.2.5. this can be entered into a section of the spreadsheet directly below the area which contains our solved deflections.

	A	B	C	D	E	F	G	H	I	J
1	**Finite difference solution of square slabs using a 5x5 mesh**									
2										
3			Grid spacing: h =	1.20	m			Uniform load: q =	15.00	kPa
4			Slab thickness: t =	0.30	m			Grid point load: P =	21.60	kN
5			Young's modulus: E =	27.00	GPa			Slab Stiffness: D =	62148	kN-m
6			Poisson's ratio: µ =	0.15				Ph^2/D =	0.5005	mm
7								D/h^2 =	43.159	kN/mm
8										
9		**A: DEFLECTIONS:w (mm)** number of iterations/10000 =							0.0391	
10										
11				*-0.473*	*-0.737*					
12			0	0	0	0	0	0		
13		*-0.473*	0	**0.4727**	**0.7368**	0.7368	0.4727	0		
14			0	0.7368	**1.1539**	1.1539	0.7368	0		
15			0	0.7368	1.1539	1.1539	0.7368	0		
16			0	0.4727	0.7368	0.7368	0.4727	0		
17			0	0	0	0	0	0		
18										
19										
20		**B: Moments: mx (kN-m/m)**								
21										
22			0	0	0	0	0	0		
23			0	**10.35**	**13.47**	13.47	10.35	0		
24			0	**15.51**	**20.70**	20.70	15.51	0		
25			0	15.51	20.70	20.70	15.51	0		
26			0	10.35	13.47	13.47	10.35	0		
27			0	0	0	0	0	0		
28										

Figure 11: Solution of a 6m x 6m simply supported plate

The moment operator formula placed in cell D23 is as follows:

"=(2*(1+$D$6)*D13-(C13+E13+$D$6*(D12+D14)))*$I$7"

Notice the calculation of D/h^2 in cell "I7". This was referenced by each of the four cells containing our moment operator. Both μ and D/h^2 were given fixed references ($I$7, and $D$6). The rest of the formula (relatively) referenced the deflections. This enabled the formula in D23 to be copied to the other three cells in the north-east quadrant of the plate. The other three quadrants were reflected by symmetry.

The calculated moments came to within 3.5% of Navier's solution given in section 4.3

We now have a useful spreadsheet model which can easily calculate the elastic deflections and moments in any simply supported square plate. It is easy to set up any of the other force operators given in section 2.2.5 if we want to to know shears, twisting moments, or edge reactions. We can also change any parameter so that for instance we could explore the effect of setting $\mu = 0$, h = 2.0 m, and t = 0.5 m:

	A	B	C	D	E	F	G	H	I	J
1		**Finite difference solution of a square slab using a 5x5 mesh**								
2										
3		Grid spacing: h =		2.00	m		Uniform load: q =		15.00	kPa
4		Slab thickness: t =		0.50	m		Grid point load: P =		60.00	kN
5	Young's modulus: E =			27.00	GPa		Slab Stiffness: D =		281250	kN-m
6		Poisson's ratio: μ =		0.00				Ph^2/D =	0.8533	mm
7								D/h^2 =	70.313	kN/mm
8										
9		**A: DEFLECTIONS: w (mm)**				number of iterations/10000 =			0.0233	
10										
11				*-0.806*	*-1.256*					
12			0	0	0	0	0	0		
13		*-0.806*	0	**0.8059**	**1.2562**	1.2562	0.8059	0		
14			0	1.2562	**1.9673**	1.9673	1.2562	0		
15			0	1.2562	1.9673	1.9673	1.2562	0		
16			0	0.8059	1.2562	1.2562	0.8059	0		
17			0	0	0	0	0	0		
18										
19										
20		**B: Moments: mx (kN-m/m)**								
21										
22			0	0	0	0	0	0		
23			0	**25.00**	**31.67**	31.67	25.00	0		
24			0	**38.33**	**50.00**	50.00	38.33	0		
25			0	38.33	50.00	50.00	38.33	0		
26			0	25.00	31.67	31.67	25.00	0		
27			0	0	0	0	0	0		

Figure 12: solution of 10m x 10m simply supported plate

4.6 Mesh Refinement

Section 4.4 postulated that the results of a finite difference analysis become more accurate as the mesh is refined. In order to examine this assertion, the 6 m by 6 m simply supported slab was iteratively solved to four decimal places for mesh sizes ranging from 2x2 to 15x15. The resulting deflections and moments at (or nearest to) the center of the slab are summarized in the following table:

Finite difference solutions of a simply supported 6m x 6m slab under uniformly distributed load

D = 62.148 kN-mm q = 15 kPa Iterations to 0.0001

Grid Size	Number of un-knowns	Time to Solve (sec)	Number of Iterations	Location * x (m)	y (m)	Results of calculation Deflection (mm)	Moment (kN-m/m)	Navier's Solution Deflection (mm)	Moment kNm/m	Percent Error Deflctn	Moment
2x2	1	<1	8	3.00	3.00	1.2219	19.406	1.2707	22.913	4.0%	18.1%
3x3	1	2	36	2.00	2.00	0.9651	17.244	0.9743	18.765	0.9%	8.8%
4x4	3	6	103	3.00	3.00	1.2600	21.831	1.2707	22.913	0.9%	5.0%
5x5	3	8	145	2.40	2.40	1.1517	20.659	1.1585	21.306	0.6%	3.1%
6x6	6	19	247	3.00	3.00	1.2621	22.315	1.2707	22.913	0.7%	2.7%
7x7	6	29	393	2.57	2.57	1.2024	21.598	1.2127	22.063	0.9%	2.2%
8x8	10	64	584	3.00	3.00	1.2558	22.364	1.2707	22.913	1.2%	2.5%
9x9	10	101	823	2.67	2.67	1.2141	21.809	1.2354	22.389	1.8%	2.7%
10x10	15	171	1114	3.00	3.00	1.2398	22.137	1.2707	22.913	2.5%	3.5%
11x11	15	225	1447	2.73	2.73	1.2033	21.621	1.2470	22.559	3.6%	4.3%
12x12	21	409	1832	3.00	3.00	1.2102	21.612	1.2707	22.913	5.0%	6.0%
13x13	21	502	2239	2.77	2.77	1.1716	20.977	1.2537	22.658	7.0%	8.0%
14x14	28	699	2687	3.00	3.00	1.1618	20.701	1.2707	22.913	9.4%	10.7%
15x15	28	886	3122	2.80	2.80	1.1155	19.887	1.2580	22.721	12.8%	14.3%

* Closest point to the centre of the plate

Table 1: Iterative solutions to 0.0001 convergence accuracy using various mesh refinements

The accuracy of the solutions is shown graphically as follows:

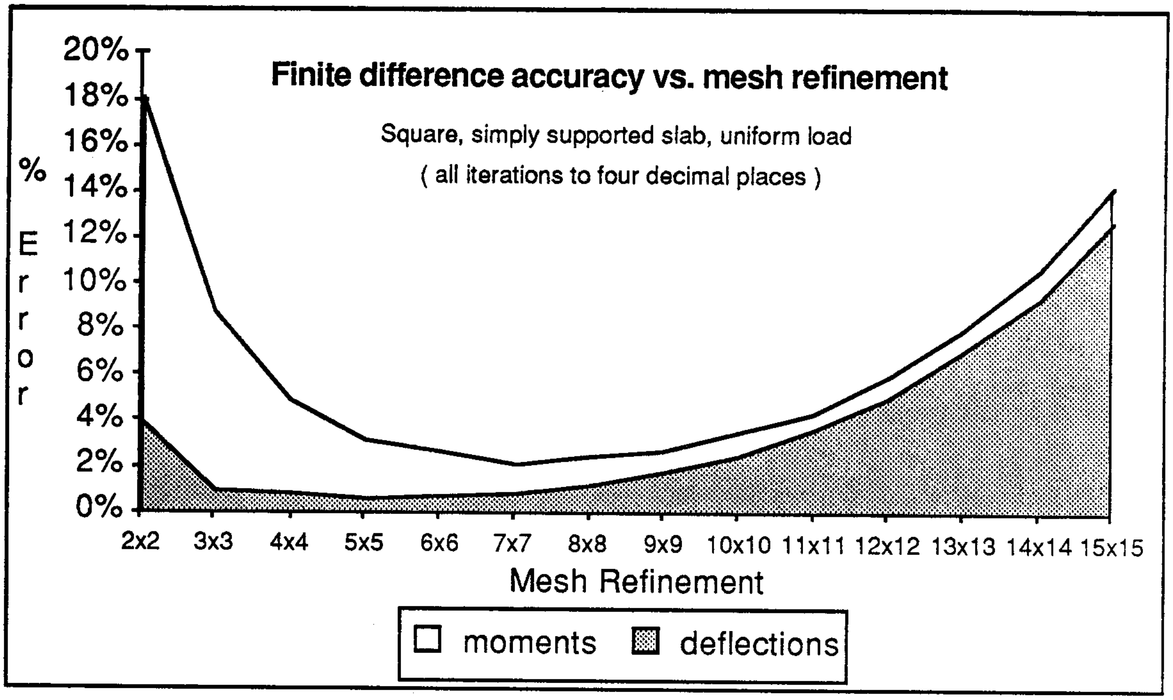


Figure 13: Finite difference accuracy vs. mesh refinement

Initially, the accuracy improved as the mesh size was decreased. However, as the mesh refinement was increased above 7x7, the accuracy started to get worse. What went wrong?

If we look back at the 5x5 mesh anlalysis from section 4.5, recall that the initial solution failed because the change between subsequent iterations was less than the 0.1 decimal accuracy specified for the completion. The correct solution for the midspan deflection was over 10 times this amount, and needed 94 iterations (with a 0.001 completion test) to approach 0.1 % accuracy. None of the 94 iterations changed the deflections more than 0.025 so the iteration terminated prematurely.

The mesh refinement study shows that the number of iterations (and time to solve the problem) increases as the mesh gets finer. If we look at the number of unknowns instead of the mesh size, two iteration values were recorded for each number of unknowns. Note that the odd mesh sizes require more iterations than the next lower (even) mesh size.

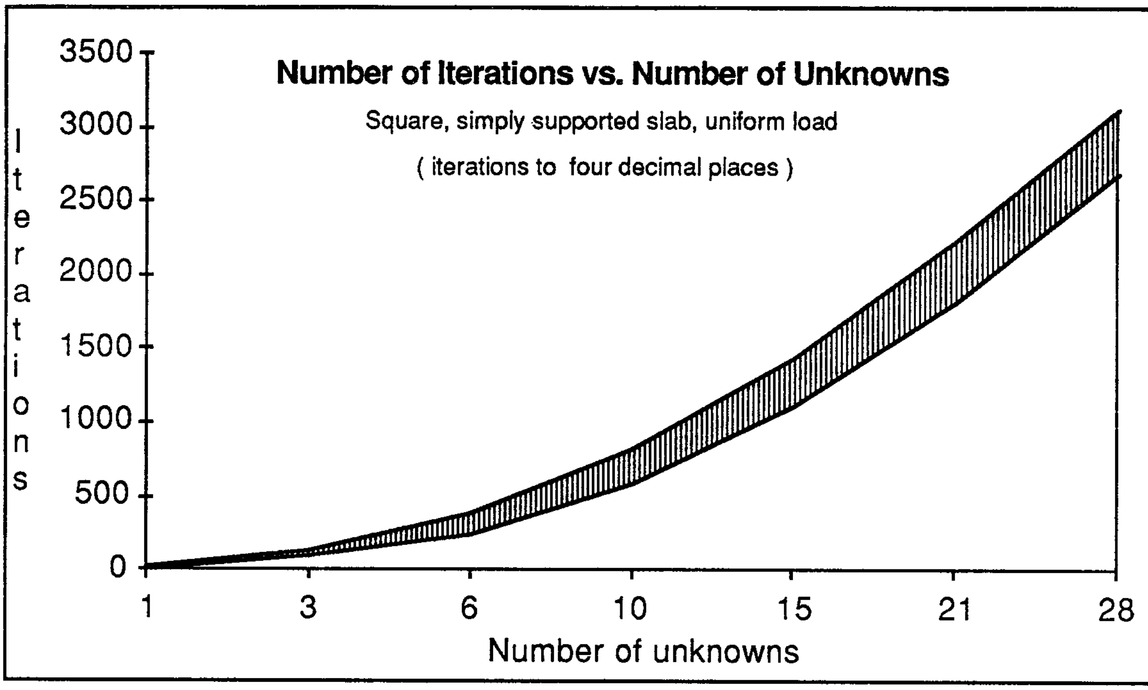


Figure 14: Number of iterations vs. number of unknowns

As the number of unknowns increases (ie. using a finer mesh), the required number of iterations increases exponentially. The changes between each iteration must be smaller if more iterations are required to arrive at the same number. Therefore the iteration will halt at lower overall precision for higher numbers of unknowns even though the same completion test is used.

If we recalculate the finer mesh sizes using a tighter completion test (0.00001 change between iterations) the results agree with the postulation that finer meshes give better accuracy:

Finite difference solutions of a simply supported 6m x 6m slab under uniformly distributed load

D = 62.148 kN-mm   q = 15 kPa   Iterations to 0.00001

Grid Size	Number of un-knowns	Time to Solve (sec)	Number of of Iterations	Location * x (m)	Location * y (m)	Results of calculation Deflection (mm)	Results of calculation Moment (kN-m/m)	Navier's Solution Deflection (mm)	Navier's Solution Moment kNm/m	Percent Error Deflctn	Percent Error Moment
12x12	21	1160	5200	3.00	3.00	1.2695	22.748	1.2707	22.913	0.1%	0.7%
13x13	21	1400	6251	2.77	2.77	1.2523	22.530	1.2537	22.658	0.1%	0.6%
14x14	28	1348	5184	3.00	3.00	1.2592	22.576	1.2707	22.9135	0.9%	1.5%
15x15	28	1815	6390	2.80	2.80	1.2431	22.350	1.2580	22.721	1.2%	1.7%

* Closest point to the centre of the plate

Table 2: Iterative solutions to 0.00001 convergence accuracy using various mesh refinements

Which then plots with the 0.0001 completion test iterations as follows:

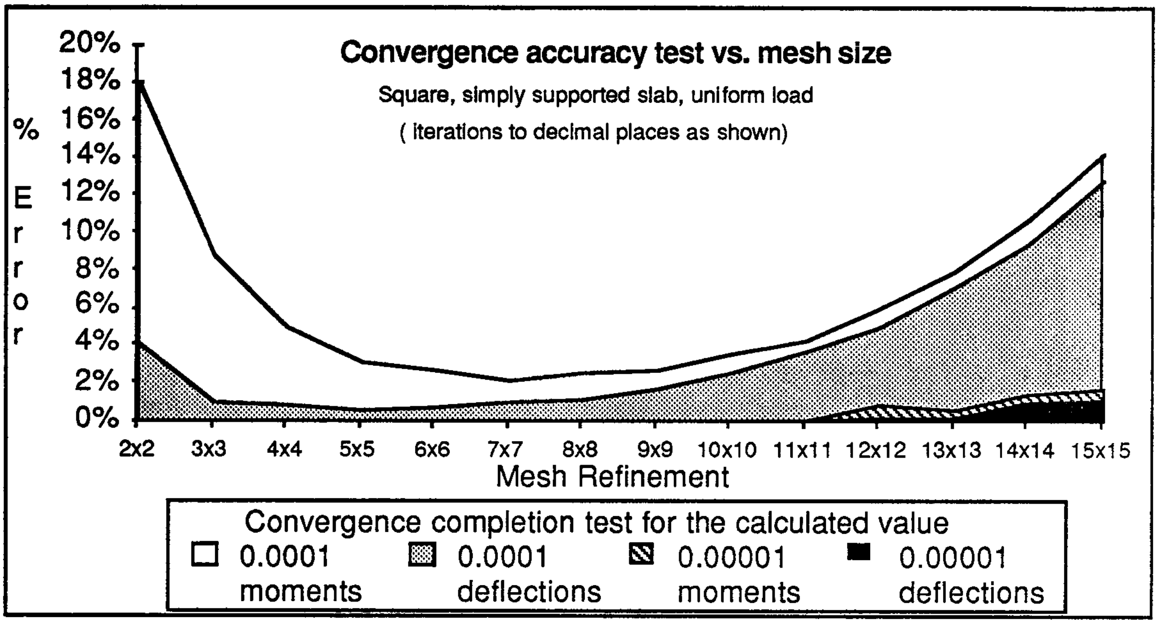


Figure 15: Convergence accuracy test vs. mesh size

This confirms the assertion that finer mesh sizes give greater accuracy. Note that it also shows that the iteration convergence test required to get at the greater accuracy must be reduced (and the number of iterations will increase).

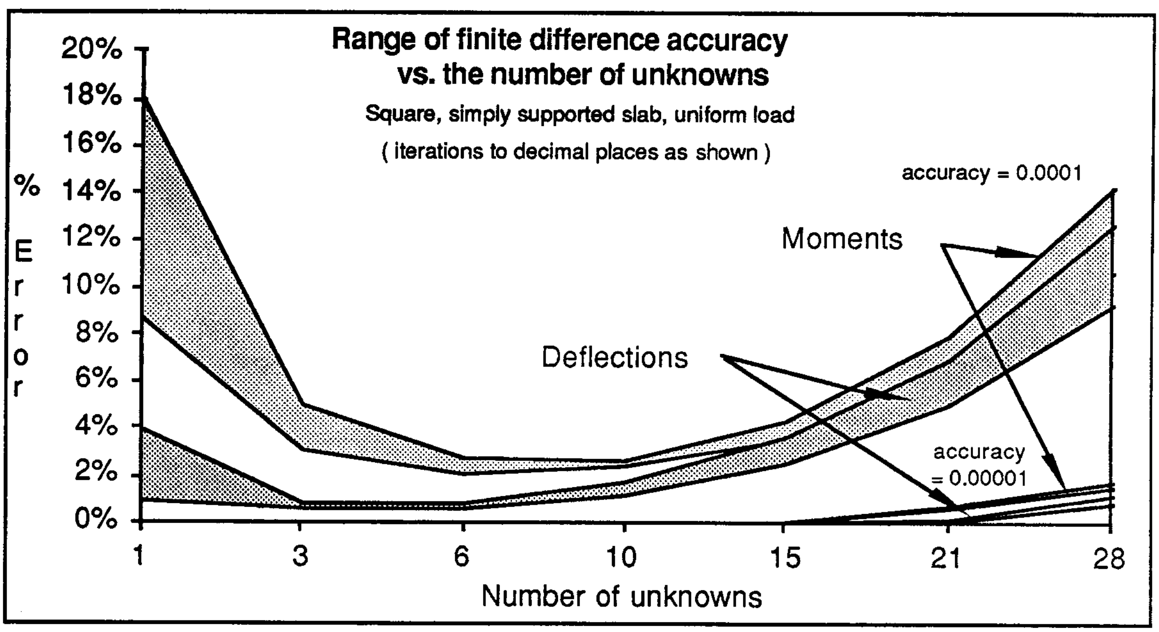


Figure 16: Range of finite difference accuracy vs. the number of unknowns

As a final point, it is worth considering how many unknowns can be practically solved using the iterative spreadsheet technique. The 0.0001 completion test gave better than 5% accuracy between 3 and 15 unknowns. This is a reasonable accuracy for many engineering problems. The amount of time that a user is willing to wait for the iterations establishes the number of unknowns and final accuracy that this method can practically be used for:

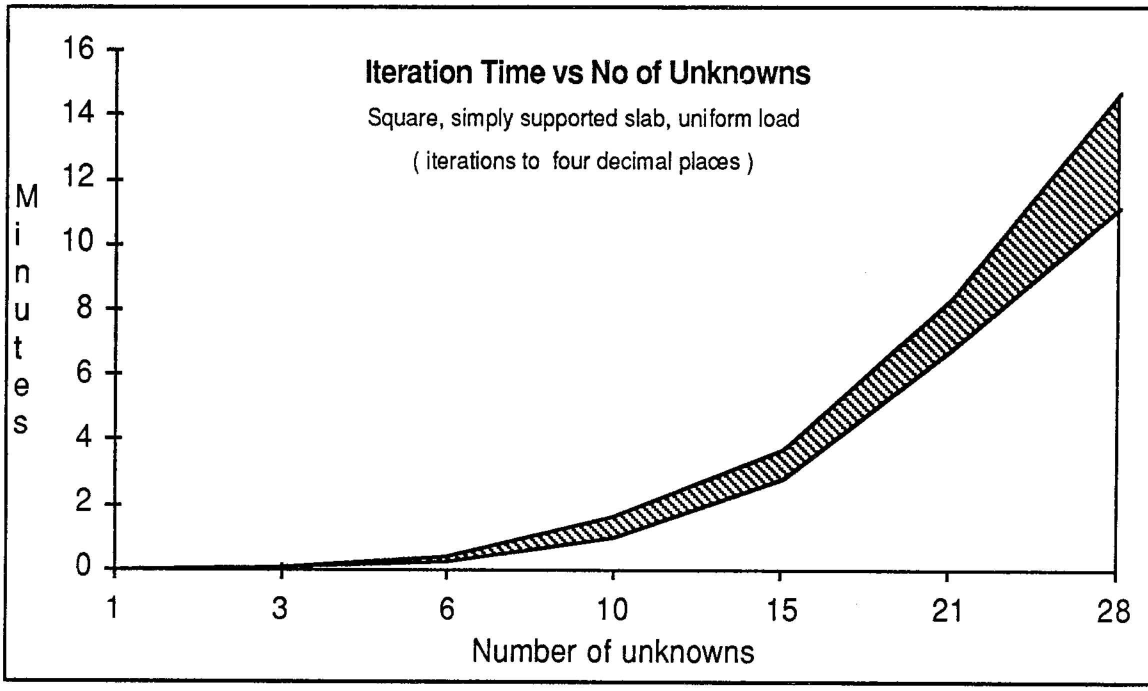


Figure 17: Iteration time vs. number of unknowns

As a further step in refining the accuracy, the 15 by 15 mesh was iterated using a completion test of 0.000001.

Finite difference solutions of a simply supported 6m x 6m slab under uniformly distributed load

D = 62.148 kN-mm   q = 15 kPa   Iterations to 0.000001

Grid Size	Number of un-knowns	Time to Solve (sec)	Number of Iterations	Location * x (m)	Location * y (m)	Results of calculation Deflection (mm)	Results of calculation Moment (kN-m/m)	Fourrier Solution Deflection (mm)	Fourrier Solution Moment kNm/m	Percent Error Deflctn	Percent Error Moment
15x15	28	2743	9659	2.80	2.80	1.2559	22.596	1.2579	22.701	0.2%	0.5%

* Closest point to the centre of the plate

Table 3: Iterative solutions to 0.000001 convergence accuracy using 15x15 mesh refinement

This iteration took almost 46 minutes. About 30 unknown deflections iterated to six place completion (giving better than 1% final accuracy) appears to be the practical limit for the author's MacIntosh® computer to solve in under one hour.

5. SOLUTION OF A FIXED EDGE PLATE WITH AN INTERIOR BEAM

5.1 Objectives

This chapter will demonstrate the spreadsheet/iteration technique for modelling a square plate with fixed edges and an interior beam. The formulation will also show how non uniform loading is incorporated into the model.

5.2 A square, fixed edge plate.

A 6m by 6m square plate with all edges fixed carries a 100 kN/m line load directly over a supporting beam:

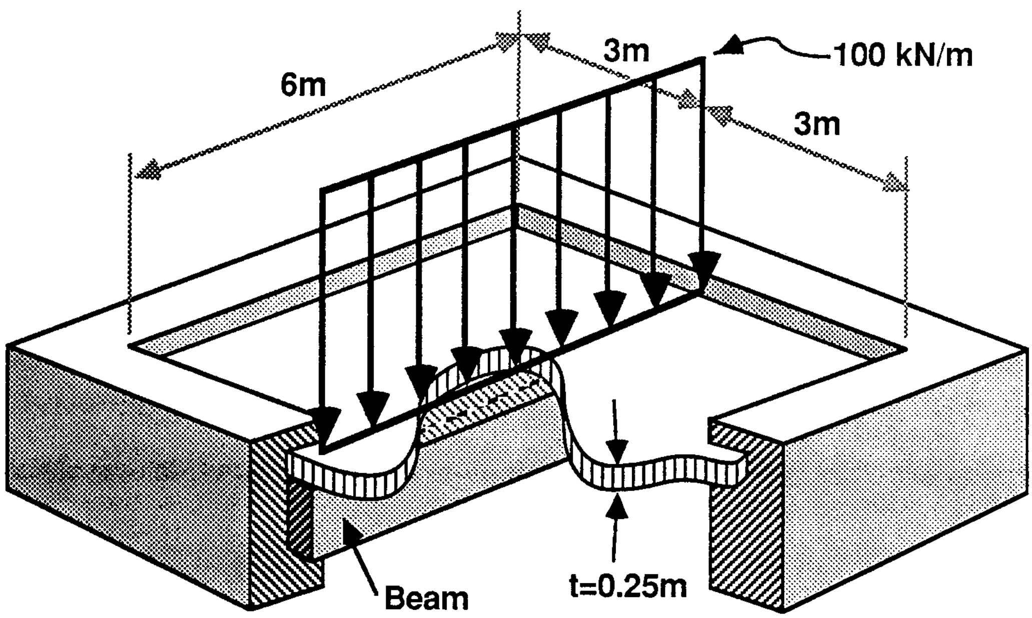


Figure 18: Square, fixed edge plate with line load and center beam

The structure is constructed out of concrete with the following properties:

Young's modulus	= E	= 27 GPa
Poisson's ratio	= μ	= 0.15

The beam is directly under the line load and has the following stiffness:

Moment of Inertia of the beam	= I	= 0.01 m^4

5.3 Solution of the primary structure

We will first solve the slab for deflections under the line load. This structure will become the primary structure on which we will superimpose the effects of the interior beam.

5.3.1 Initialization of the spreadsheet model

As in section 4.5, the problem parameters should be entered near the top of the spreadsheet and appropriately labelled. Equations should be set up to calculate constants such as the mesh spacing, D, EI, h^2/D, etc. Changes to the parameters such as the slab thickness will then automatically update these constants. The beam Inertia and calculated stiffness constants were entered for use in section 5.4 when we add the beam to the primary structure.

	A	B	C	D	E	F	G	H	I	J
1	Solution of a square, fixed edge slab with an interior beam under a line load. 6X6 mesh size									
2										
3			Span: L =	6.000	m		Grid spacing: h =	1.000	m	
4			Slab thickness: t =	0.250	m		Grid point load: P =	100.0	kN	
5			Young's modulus: E =	27.000	GPa		Slab stiffness: D =	35965	kN-m	
6			Poisson's ratio: μ =	0.150			Beam stiffness: EI =	270000	kN-m^2	
7			Beam moment of inertia: I =	0.010	m^4		h^2/D =	0.0278	mm/kN	
8			Line load on slab: W =	100	kN/m		D/h^2 =	35.97	kN/mm	
9							EI/h^3 =	270.00	kN/mm	
10			Iterations/100000 =	0.00266			2+2μ =	2.300		

Note that all values in column "H" are calculated from the parameters entered in column "D". As was done in the last chapter, a function to keep track of the iterations was entered in cell "D10".

5.3.2 Point loads for each node on the mesh

Since the load is not uniform over the whole slab, a section of the spreadsheet will be required to map the values of the point loads on each node:

	A	B	C	D	E	F	G	H
11								
12				Point loads on the plate (kN)				
13								
14			0	0	100	0	0	
15			0	0	100	0	0	
16			0	0	100	0	0	
17			0	0	100	0	0	
18			0	0	100	0	0	
19								

The values in cells E14 to E18 all reference the value calculated in cell H4 so that if we change the line loading it will automatically be carried through the rest of the spreadsheet.

5.3.3 Entry of the plate and fictitious point operators

The region of the spreadsheet where the plate operators are entered is set up as follows:

	A	B	C	D	E	F	G	H
21				Node deflections: w (mm)				
22			*0.0000*	*0.0000*	*0.1390*			
23		0	0	0	0	0	0	0
24	*0.0000*	0	**0.0000**	**0.0000**	**0.1390**	0.0000	0.0000	0
25	*0.0000*	0	**0.0000**	**-0.0139**	**0.1891**	-0.0139	0.0000	0
26	*0.0014*	0	**0.0014**	**-0.0239**	**0.2627**	-0.0239	0.0014	0
27		0	0.0000	-0.0139	0.1891	-0.0139	0.0000	0
28		0	0.0000	0.0000	0.1390	0.0000	0.0000	0
29		0	0	0	0	0	0	0

The equation entered into cell E26 is the plate operator developed in section 2.2.4 :

"=($H$7*E16-(E24+G26+E28+C26+2*(D25+F25+F27+D27)
-8*(E25+F26+E27+D26)))/20"

Note the use of fixed reference ($H$7) for the value h^2/D which is then multiplied by the relatively referenced (E16) point load "P" to get Ph^2/D. This formula was copied to cells C24 through E26 and correctly references the values of "P" stored in cells C14 to E16. As in section 4.5, the spreadsheet was set up to perform iterations only when requested.

The line load in the north-south direction gives the problem only orthogonal symmetry so that nine values of the deflection must be solved. (Recall that the simply supported slab analysed in section 4.5 also had diagonal symmmetry so that only six deflections were solved when the same mesh was used.)

The fictitious points were set equal to the deflections on the opposite side of the plate boundary in accordance with the fixed edge operator developed in section 2.3.3.2.

5.3.4 Iterate to a solution

	A	B	C	D	E	F	G	H
21				Node deflections: w (mm)				
22			*0.1743*	*0.5202*	*0.8330*			
23		0	0	0	0	0	0	0
24	*0.1743*	0	**0.1743**	**0.5202**	**0.8330**	0.5202	0.1743	0
25	*0.3983*	0	**0.3983**	**1.0958**	**1.6312**	1.0958	0.3983	0
26	*0.4892*	0	**0.4892**	**1.3202**	**1.9297**	1.3202	0.4892	0
27		0	0.3983	1.0958	1.6312	1.0958	0.3983	0
28		0	0.1743	0.5202	0.8330	0.5202	0.1743	0
29		0	0	0	0	0	0	0

The above solution required 168 iterations to achieve convergence to less than 0.0001 decimal change between subsequent iterations.

5.3.5 Calculate the bending moments

The operators for m_{OX} and m_{OY} from section 2.2.5 were entered to calculate the bending moments throughout the north-west quadrant of the slab. This completed the solution of the primary structure:

	A	B	C	D	E	F	G	H	I	J
1	Solution of a square, fixed edge slab with an interior beam under a line load. 6X6 mesh size									
2										
3			Span: L =	6.000	m		Grid spacing: h =	1.000	m	
4			Slab thickness: t =	0.250	m		Grid point load: P =	100.0	kN	
5			Young's modulus: E =	27.000	GPa		Slab stiffness: D =	35965	kN-m	
6			Poisson's ratio: μ =	0.150			Beam stiffness: EI =	270000	kN-m^2	
7			Beam moment of inertia: I =	0.010	m^4		h^2/D =	0.0278	mm/kN	
8			Line load on slab: W =	100	kN/m		D/h^2 =	35.97	kN/mm	
9							EI/h^3 =	270.00	kN/mm	
10			Iterations/100000 =	0.00170			2+2μ =	2.300		
11										
12				Point loads on the plate (kN)						
13										
14			0	0	100	0	0			
15			0	0	100	0	0			
16			0	0	100	0	0			
17			0	0	100	0	0			
18			0	0	100	0	0			
19										
20										
21				Node deflections: w (mm)						
22			*0.1743*	*0.5202*	*0.8330*					
23		0	0	0	0	0	0	0		
24	*0.1743*	0	**0.1743**	**0.5202**	**0.8330**	0.5202	0.1743	0		
25	*0.3983*	0	**0.3983**	**1.0958**	**1.6312**	1.0958	0.3983	0		
26	*0.4892*	0	**0.4892**	**1.3202**	**1.9297**	1.3202	0.4892	0		
27		0	0.3983	1.0958	1.6312	1.0958	0.3983	0		
28		0	0.1743	0.5202	0.8330	0.5202	0.1743	0		
29		0	0	0	0	0	0	0		
30										
31				Moments mx (kN-m/m)						
32		0.00	-1.88	-5.61	-8.99					
33		-12.54	-6.44	0.90	22.68					
34		-28.65	-10.05	7.72	41.21					
35		-35.19	-11.32	10.39	47.06					
36										
37										
38										
39										
40				Moments my (kN-m/m)						
41		0.00	-12.54	-37.42	-59.92					
42		-1.88	-2.71	-1.81	4.62					
43		-4.30	3.17	13.50	23.75					
44		-5.28	4.69	17.34	28.04					
45										

Figure 19: Spreadsheet solution of the fixed edge slab under line loading

5.4 Superposition of the center beam

Section 2.4.1.2 developed an operator to calculate the loads required to hold the beam in the shape dictated by an array of deflections {w}. If we apply this operator to the deflections of the slab which are directly over the beam, we can calculate the forces required to hold the beam in the same deflected shape. These reactions from the beam can then be applied as loads on the slab. Applying these loads we will calculate a new deflected shape for the slab which gives a new set of beam reactions. If we continue to iterate, applying each new set of beam reactions to the slab we **MAY** converge to a solution. Special steps are outlined in section 5.4.3 to ensure that the solution is reached.

5.4.1 Calculation of beam reaction forces

The next step in the solution is to calculate the forces required to hold the beam to the same deflections as the slab.

	A	B	C	D	E	F	G	H	I	J
11										Beam
12				Point loads on the plate (kN)						Reaction
13										(kN)
14			0	0	100	0	0			333.6
15			0	0	100	0	0			99.4
16			0	0	100	0	0			52.4
17			0	0	100	0	0			
18			0	0	100	0	0			
19										
20										
21				Node deflections: w (mm)						
22			*0.1743*	*0.5202*	*0.8330*					
23		0	0	0	0	0	0	0		
24	*0.1743*	0	**0.1743**	**0.5202**	**0.8330**	0.5202	0.1743	0		
25	*0.3983*	0	**0.3983**	**1.0958**	**1.6313**	1.0958	0.3983	0		
26	*0.4892*	0	**0.4892**	**1.3203**	**1.9297**	1.3203	0.4892	0		
27		0	0.3983	1.0958	1.6313	1.0958	0.3983	0		
28		0	0.1743	0.5202	0.8330	0.5202	0.1743	0		
29		0	0	0	0	0	0	0		

The equation entered into cell J16 was the beam force/deflection operator from section 2.4.1.2 which calculates the force required to hold the beam in the same deflected shape as the slab:

"=$H$9*(6*E26-4*(E25+E27)+E24+E28)"

This was subsequently copied to cells J14 and J15 to calculate their forces.

5.4.2 Addition of the beam reaction forces to the slab

Sum the calculated beam reaction forces (in cells J14 to J16) with the line loads (in cells E14 to E16) by modifying the plate operators in cells E24 to E26.

	A	B	C	D	E	F	G	H
21				Node deflections: w (mm)				
22			0.1743	0.5202	0.8330			
23		0	0	0	0	0	0	0
24	0.1743	0	0.1743	0.5202	-0.0576	0.5202	0.1743	0
25	0.3983	0	0.3983	1.0958	0.5127	1.0958	0.3983	0
26	0.4892	0	0.4892	1.3203	1.4539	1.3203	0.4892	0
27		0	0.3983	1.0958	0.5127	1.0958	0.3983	0
28		0	0.1743	0.5202	-0.0576	0.5202	0.1743	0
29		0	0	0	0	0	0	0

The typical modifications to cell E26 are highlighted in following equation:

"=($H$7*(E16-J16)-(E24+G26+E28+C26+2*(D25+F25+F27+D27)-8*(E25+F26+E27+D26)))/20"

5.4.3 The need for a smoothing function

When the above spreadsheet was iterated **the values did not converge to a solution.** The beam forces started to oscillate wildly between increasing positive and negative values.

If the reader wishes to observe this phenomenon he should make a backup copy of his spreadsheet before commencing the iterations. Most spreadsheet programs do not offer any means to reset the deflected values once they start to diverge . When the formulas are modified to correct a divergence problem, the spreadsheet must iterate to the correct solution from the diverged values. This process takes considerably longer than erasing and reconstructing the plate and beam operator formulas. The fastest way to recover is to open a backup copy of the worksheet which was saved before the diverging iterations occurred.

If you save a backup copy of your spreadsheet and iterate, you will notice that the oscillation of the slab-beam deflections is not a fundamental displacement mode. Some nodes will deflect upwards as others deflect downwards.

The reason that the solution diverged is that the beam "over reacted" (pardon the pun) to the deflections imposed by the plate. The stiffness of the beam is much greater than the stiffness of the slab so it takes large forces to force it to the slab's deflected shape. See how the reaction force displayed in cell J14 in section 5.4.1 is much greater than the applied line load. In other words, the reaction force calculated in the first iteration will locally deflect the slab upwards in the next iteration. When the slab is locally deflected upwards, the beam goes into negative curvature over that node and will try to pull the slab back down. The reaction will then sum with the downward line load so that, after the next iteration, the slab will deflect downward further than before the beam was added to the structure. Subsequent iterations will cause greater and greater upward and downward deflections to occur.

Some sort of smoothing function is needed to damp out these diverging oscillations and yet allow the slab and beam to converge to compatible deflections.

The function must reduce the beam's reaction forces initially, when they are fluctuating wildly, but increase them to their full value as the solution converges. A reasonable approach might be to average the beam reaction forces between consecutive iterations.

The following formula was used to get a smoothed value of the reaction after the i^{th} iteration :

$$R_{(i)} = R\{w\}_{(i)}/2 + R_{(i-1)}/2$$

where:

$R_{(i)}$	is the smoothed value of the reaction
$R\{w\}_{(i)}$	is the unsmoothed reaction
$R_{(i-1)}$	is the smoothed value from the last iteration

While this function works well for small values of beam stiffness, it does not work when the beam stiffness is considerably greater than the slab stiffness. It was therefore proposed that the smoothing function incorporate the ratio of beam to slab stiffness as follows:

$$R_{(i)} = R\{w\}_{(i)}/S + R_{(i-1)}*(S-1)/S$$

where: "S" is called the smoothing factor and is set equal to EI/Dh

In the very first iteration, the value of "$R_{(i-1)}$" is equal to zero. Therefore the value of "$R_{(i)}$" equals "$R\{w\}_{(i)}/S$" which is only $1/S^{th}$ of the true reaction for the deflected shape of the slab. If "R{w}" changes drastically in the next iteration, as a result of the previous "$R_{(i)}$" forcing the slab into a new shape, it will be proportionally averaged with this "$R_{(i)}$" value. Consequently, the new reaction pushes the slab to the required deflection a little more gently. As the spreadsheet iterates towards the true solution , "$R_{(i)}$" will change very little between subsequent iterations and becomes almost equal to "$R_{(i-1)}$". When convergence is finally reached, "R{w}" = "$R_{(i)}$" = "$R_{(i-1)}$".

5.4.4 Formulation of the smoothing function

Equations for the beam reactions were corrected to include the smoothing function as follows:

	A	B	C	D	E	F	G	H	I	J
8	Line load on slab: W =			100	kN/m		D/h^2 =	35.97	kN/mm	
9	Smoothing factor: S =			7.5072	(S-1)/S =	0.867	EI/h^3 =	270.00	kN/mm	
10	Iterations/100000 =			0.00170			2+2μ =	2.300		
11										Beam
12				Point loads on the plate (kN)						Reaction
13										(kN)
14			0	0	100	0	0		334	333.6
15			0	0	100	0	0		99	99.4
16			0	0	100	0	0		52	52.4
17			0	0	100	0	0			
18			0	0	100	0	0			

The smoothing factor "S" in cell D9 was entered on the spreadsheet as the ratio of the beam and slab stiffnesses in cells I9 and I8. Thus if we change the value of the beam moment of inertia, slab thickness, or Poisson's ratio, the smoothing function will automatically be adjusted to suit. The ratio (S-1)/S was placed in cell F9 for use by the smoothing function.

Cells I14 to I16 were set equal to the values in cells J14 to J16. After each iteration they will contain the value of the smoothed beam reaction from the previous iteration.

The formulas for beam reactions in cell J14 to J16 were modified to incorporate the smoothing function. The changes to the formula in J16 are highlighted below:

"=$H$9*(6*E26-4*(E25+E27)+E24+E28)**/$D$9+I16*$F$9**"

Thus the smoothing function was very easily incorporated into our spreadsheet model. Much easier to do, in fact, than to explain in section 5.4.3.

5.4.5 <u>Iterate to the solution</u>

The rest is easy, we simply start the spreadsheet iterating and wait for it to converge to the solution:

	A	B	C	D	E	F	G	H	I	J
1	Solution of a square, fixed edge slab with an interior beam under a line load. 6X6 mesh size									
2										
3			Span: L =	6.000	m	Grid spacing: h =		1.000	m	
4			Slab thickness: t =	0.250	m	Grid point load: P =		100.0	kN	
5			Young's modulus: E =	27.000	GPa	Slab stiffness: D =		35965	kN-m	
6			Poisson's ratio: μ =	0.150			Beam stiffness: EI =	270000	kN-m^2	
7			Beam moment of inertia: I =	0.010	m^4		h^2/D =	0.0278	mm/kN	
8			Line load on slab: W =	100	kN/m		D/h^2 =	35.97	kN/mm	
9			Smoothing factor: S =	7.5072	(S-1)/S =	0.867	EI/h^3 =	270.00	kN/mm	
10			Iterations/100000 =	0.00266			2+2μ =	2.300		
11										Beam
12				Point loads on the plate (kN)						Reaction
13										(kN)
14			0	0	100	0	0		*77*	76.6
15			0	0	100	0	0		*54*	54.4
16			0	0	100	0	0		*47*	47.3
17			0	0	100	0	0			
18			0	0	100	0	0			
19										
20										
21				Node deflections: w (mm)						
22			*0.0706*	*0.2070*	*0.3183*					
23		0	0	0	0	0	0	0		
24	*0.0706*	0	**0.0706**	**0.2070**	**0.3183**	0.2070	0.0706	0		
25	*0.1716*	0	**0.1716**	**0.4709**	**0.7003**	0.4709	0.1716	0		
26	*0.2149*	0	**0.2149**	**0.5813**	**0.8568**	0.5813	0.2149	0		
27		0	0.1716	0.4709	0.7003	0.4709	0.1716	0		
28		0	0.0706	0.2070	0.3183	0.2070	0.0706	0		
29		0	0	0	0	0	0	0		
30										
31				Moments mx (kN-m/m)						
32		0.00	-0.76	-2.23	-3.43					
33		-5.08	-2.53	0.59	7.66					
34		-12.34	-4.28	3.34	17.72					
35		-15.46	-4.98	4.46	21.51					
36										
37										
38										
39										
40				Moments my (kN-m/m)						
41		0.00	-5.08	-14.89	-22.89					
42		-0.76	-1.45	-1.91	-1.09					
43		-1.85	1.38	5.90	10.58					
44		-2.32	2.30	8.43	14.23					
45										

Figure 20: Spreadsheet solution of the fixed edge slab with a beam

5.4.6 Enhancements to the spreadsheet model

We can easily modify our spreadsheet to model other types of structures. Some of the possiblilites are given below:

- The input parameters in cells D3 to D8 can be changed as required to see their effect on the solution of the problem.
- The loading on the slab can be changed by modifying the values in cells C14 to G18. If the loading is not symmetric, the deflections won't be either so the typical plate operator (in cell C24 for example) can be copied to any other cells (nodes) where the deflection are no longer known by symmetry. Care must be taken that deflections for fictitious points are properly entered where they are required by the operators.
- If the beam has torsional stiffness it can be incorporated by using the operator developed in section 2.4.2. A smoothing function may also be needed to prevent divergence of the solution. Interested readers may wish to develop an operator to calculate the distribution of the moment on each side of the beam since the moment operator will only give an average value of m_X over the beam (you might follow the approach used to get the operator for M_{OY} in section 2.4.2).
- Adding a negative sign to the formula for the fictitious points will make the edge simply supported. You can make some of the edges fixed and some simply supported but must be careful to copy the plate operators to all nodes whose deflections are no longer known by symmetry.

6. SOLUTION OF A RECTANGULAR PLATE WITH EDGE BEAMS

6.1 Objectives

This chapter will demonstrate the spreadsheet/iteration technique for modelling a rectangular plate with two free edges and two edges supported by torsionally stiff beams. Uniform loading and the self weight of the beams will be used so that the problem is doubly symmetric.

6.2 A rectangular plate with two free edges and two edge beams

A 6m by 4m slab with two edge beams is simply supported at all four corners:

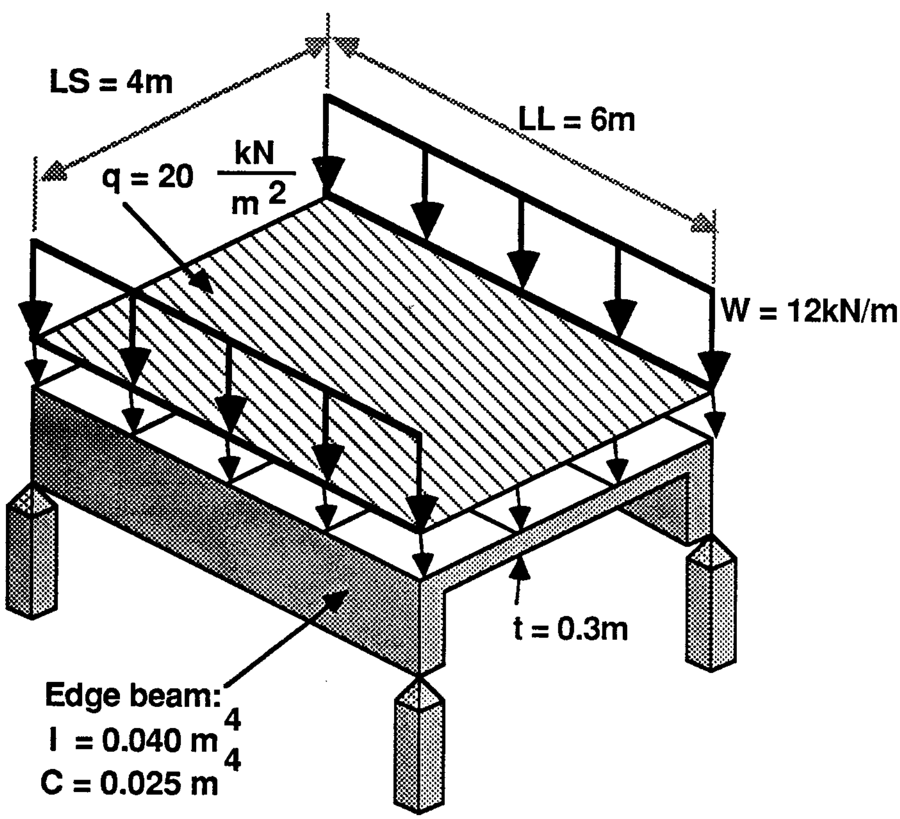


Figure 21: Rectangular plate with edge beams

The following properties were assumed for the concrete:

Young's modulus	= E	= 29 GPa
Poisson's ratio	= μ	= 0.17

6.3 Solution of the primary structure

We will first solve a 4m x 6m free edge slab under uniform loading without including the edge beams.

6.3.1 Initialization of the spreadsheet model

As in section 5.3.1, the problem parameters were entered near the top of the spreadsheet and appropriately labeled. Equations were set up to calculate LL, h, D, EI, h^2/D, G, $CG/2h^3$, etc. so that other structures can be easily analyzed by changing the parameters in cells D4 to D11. Note that this spreadsheet model will only analyze rectangular plates where the long span is 1.5 times the short span.

	A	B	C	D	E	F	G	H	I	J
1	Solution of a rectangular plate with two free edges and two edge beams, simply supported									
2				Mesh size = 4X6						
3										
4			Short span length: LS =	4.000	m			Long span length: LL =	6.000	m
5			Slab thickness: t =	0.300	m			Grid spacing: h =	1.000	m
6			Uniformly distr. load: q =	20.0	kN/m^2			Node load due to q: Pq =	20.000	kN
7			Beam weight: W =	12.0	kN/m			Node load due to W: PW =	12.000	kN
8			Beam moment of inertia: I =	0.040	m^4			Bending stiffness: EI/h^3 =	1160.0	kN/mm
9			B'm torsional constant: C =	0.025	m^4			Torsional stiffness: CG/h^3 =	309.8	kN/mm
10			Young's Modulus: E =	29.0	GPa			Shear Modulus: G =	12.393	GPa
11			Poisson's Ratio: μ =	0.170				D =	67192	kN-m
12								h^2/D =	0.01488	mm/kN
13								2+2μ =	2.340	
14			Iterations/100000 =	0.00000				CG/2h^3 =	154.9	kN/mm
15										
16										
17	Reaction from edge beam:					Smoothing factor: Sb =	17.264		(Sb-1)/Sb =	0.94208
18										
19	R(i) =					kN				
20	R(i-1) =					kN				
21										
22	Moment from edge beam:					Smoothing factor: St =	4.61111		(St-1)/St =	0.78313
23										
24	M(i) =					kN-m/m				
25	M(i-1) =					kN-m/m				

Notice that room was set aside for the beam reactions and moments which will be added after the primary structure has been set up. The smoothing factor in cell G17 gives the product of EI/h^3 (in cell I8) and h^2/D (in cell I12) to get EI/hD. The smoothing factor for the torsional moments along the edge beam in cell G22 contains the formula "=I9*I12" to get the stiffness ratio CG/hD.

6.3.2 Point loads for each node on the mesh

Even if the self weight of the beams, W, is zero, the uniformly distributed load will not give identical loads at all nodes of the mesh. This is because the nodes on the free edge have only half the tributary area of a node at the interior of the slab (This was not considered in chapter 4 because all edges were supported so that the load on the edge node was of no consequence.) For this reason, and to include the self weight of the beams, an area of the spreadsheet was set up to calculate the loads at each of the nodes :

	A	B	C	D	E	F	G	H	I
27		LOADS AT NODES OF THE PRIMARY STRUCTURE (kN)							
28		0	22	22	22	22	22	0	
29		10	20	20	20	20	20	10	
30		10	20	20	20	20	20	10	
31		10	20	20	20	20	20	10	
32		0	22	22	22	22	22	0	

The values in cells C29 to G31 all reference the value of Pq calculated in cell I6. Cells C28 to G28 contain the value of PW in cell I7 added to half of Pq. The free edge in cells B29 to B31 only carries Pq/2. The corners were not given any load because they are support points and their loads will not affect the solution. Symmetry was used to generate the rest of the loads.

6.3.3 Entry of the plate and fictitious point equations

After setting the spreadsheet to perform iterations only when requested, the region of the spreadsheet containing the plate deflection operators was set up as follows:

	A	B	C	D	E	F	G	H	I
34				NODE DEFLECTIONS (mm)					
35		*0.000*	*0.000*	*0.000*	*0.000*	*0.000*			
36	*0.000*	0	**0.0327**	**0.0458**	**0.0494**	0.0458	0.0327	0	
37	*0.000*	**0.0086**	**0.0268**	**0.0353**	**0.0383**	0.0000	0.0000	0.0086	
38	*-0.001*	**0.0130**	**0.0249**	**0.0295**	**0.0347**	0.0000	0.0000	0.0000	
39	*0.000*	0.0086	0.0268	0.0353	0.0383	0.0000	0.0000	0.0086	
40		0	0.0327	0.0458	0.0494	0.0458	0.0327	0	

Please note that the values displayed above are very dependant on the order in which the formulas were entered and have little meaning before iterations begin. This is because each operator calculates its value as it is entered. The value displayed by one operator is used in calculating the value for any subsequent adjacent operators. If you were to create the same system of equations in the same cells, your display would likely be different because of a different formula entry sequence. During iteration however, all of the deflections will converge the same solution provided that the operators were correctly formulated.

Cells C37 to E38 contained the basic plate operator equation from section 2.2.4. For example, cell C37 contained the formula:

"(C29*$I$12-(C35+E37+C39+A37+2*(B36+D36+D38+B38)
-8*(C36+D37+C38+B37)))/20"

Cells C36 to E36 and B37 to B38 contained the free edge operator developed in section 2.3.4, rotated appropriately for north and west edges. For example, the following equation was entered into cell C36 and copied to cells D36 and E36:

"=C28*$I$12/10-($D$11*((B35+D35-B37-D37)/2+C37-C35)-7*C37
+2*(B37+D37)-4*(B36+D36)+(A36+E36)/2+C38-C35)/10"

The fictitious deflections in Cells C35 to E35 and A37 to A38 use the operator developed in section 2.3.1, appropriately rotated, with edge moments equal to zero. An example of one of these equations, is contained in cell A37:

"=$I$13*B37-$D$11*(B36+B38)-C37"

Finally, operators for simple supports (section 2.3.3.1) were placed in cells B35 and B36 which negatively mirror the deflections on the opposite side of the corner support.

6.3.4 Iterate to a solution

We now have a complete formulation of our primary structure (free edge plate, no edge beams). It isn't necessary to iterate to a solution at this time but we may want to compare the ultimate solution, with edge beams, to the free edge plate. It is also possible that we have made an error in our formulation so we could perform a few hand calculations on the solution to confirm that the operators are correct so far.

	A	B	C	D	E	F	G	H
33				NODE DEFLECTIONS (mm)				
34		*-1.303*	*3.552*	*6.813*	*7.955*	*6.813*		
35	*-3.937*	0	**3.9371**	**6.6923**	**7.6738**	6.6923	3.9371	0
36	*-1.782*	**1.3026**	**4.5230**	**6.8734**	**7.7266**	6.8734	4.5230	1.3026
37	*-1.001*	**1.8038**	**4.7791**	**6.9814**	**7.7866**	6.9814	4.7791	1.8038
38	*-1.782*	1.3026	4.5230	6.8734	7.7266	6.8734	4.5230	1.3026
39		0	3.9371	6.6923	7.6738	6.6923	3.9371	0

The above solution required 1194 iterations to reach convergence to less than 0.0001 mm change in deflections. The reader can confirm by hand calculation that moments along all edges are zero, and that the interior and free edge plate operators are correct.

6.4 Addition of beams to the structure

The edge beams can be added to the primary structure in two stages. This allows us to verify the formulation of each stage and see the effect of each additional constraint.

6.4.1 Beam flexural reaction forces

The beam force/deflection operator from section 2.4.1.2 was placed in cells C18 to E18 and included a smoothing function identical to the one used in section 5.4.4:

	A	B	C	D	E	F	G	H	I	J
16	Reaction from edge beam:			Smoothing factor: Sb =			17.264		(Sb-1)/Sb =	0.94208
17										
18	R(i) =		39.66	27.04	25.44	KN				
19	R(i-1) =		*39.66*	*27.04*	*25.44*	*KN*				

The formula was entered in cell C18 and before being copied to cells D18 and E18. The formula in cell C18 with the smoothing function highlighted for clarity is:

"=$I$8*(6*C35-4*(B35+D35)+A35+E35)<u>/$G$16+C19*$J$16</u>"

Cells C19 to E19 were set equal to the values in cells C18 to E18 as required by the smoothing function.

The reactions in cells C18 to E18 were applied to the plate by summing them with the node loads in cells C28 to E28. The free edge operator in cells C36 to E36 was modified in the same manner as in section 5.4.2.

After 425 more iterations the solution converged to the following deflections:

	A	B	C	D	E	F	G	H
33				NODE DEFLECTIONS (mm)				
34		*-0.839*	*-0.377*	*-0.071*	*0.036*	*-0.071*		
35	*-0.345*	0	**0.3450**	**0.5915**	**0.6803**	0.5915	0.3450	0
36	*0.680*	**0.8392**	**1.0840**	**1.2811**	**1.3549**	1.2811	1.0840	0.8392
37	*1.079*	**1.1733**	**1.3816**	**1.5600**	**1.6282**	1.5601	1.3816	1.1733
38	*0.680*	0.8392	1.0840	1.2811	1.3549	1.2811	1.0840	0.8392
39		0	0.3450	0.5915	0.6803	0.5915	0.3450	0

6.4.2 Beam torsional reaction forces

Beam torsional reactions at the edge of the slab were calculated using the torsional operator developed in section 2.4.2:

	A	B	C	D	E	F	G	H	I	J
21	Moment from edge beam:				Smoothing factor: St =		4.61111		(St-1)/St =	0.89157
22										
23	m(i) =	-14.584	3.626	2.547	2.238	kN-m/m				
24	m(i-1) =	0.000	0.000	0.000	0.000	*kN-m/m*				

The equation in cell C23 (copied to cells D23 and E23) includes the smoothing function highlighted below:

"=2*$I$14*(-C34+C36+(B34-B36))<u>/**$G$21+C24*$J$21**</u>"

Since we know that beam torsion cannot occur to the left of cell B23 and the fictitious deflection of node A34 is not known, the operator in cell C23 cannot be copied to cell B23. Referring to section 2.4.2 we see that when torsion only occurs to the right of a node:

$$T_{OE} = CG/(2h^2)\{W_{NE}-W_{NW}+w_{SW}-w_{SE}\}$$

The slab edge moment which must be in equilibrium with this torsion will be distributed along a length of h/2 at the corner. Therefore the value of the distributed moment becomes:

$$m_Y = 2CG/(2h^3)\{W_{NE}-W_{NW}+w_{SW}-w_{SE}\}$$

This was entered into cell B23, above, and incorporates the smoothing function:

"=$I$14*(-B34-D34+B36+D36+2*(C34-C36))<u>/**$G$21+C24*$J$21**</u>"

6.4.2.1 Modification of fictitious point operators to include torsional reactions from the beam

The edge moments calculated in cells B23 to E23 were added to the fictitious deflection equations from section 2.3.1 which were placed in cells B34 to E34:

	A	B	C	D	E	F	G	H
33				NODE DEFLECTIONS (mm)				
34		*-0.622*	*-0.431*	*-0.109*	*0.003*	*-0.109*		
35	*-0.345*	0	**0.3450**	**0.5915**	**0.6803**	0.5915	0.3450	0
36	*0.680*	**0.8392**	**1.0840**	**1.2810**	**1.3548**	1.2810	1.0840	0.8392
37	*1.079*	**1.1733**	**1.3815**	**1.5600**	**1.6282**	1.5600	1.3815	1.1733
38	*0.680*	0.8392	1.0840	1.2810	1.3548	1.2810	1.0840	0.8392
39		0	0.3450	0.5915	0.6803	0.5915	0.3450	0

B34 contains the following formula which was copied into cells C34 to E34:

"=$I$13*B35-$D$11*(A35+C35)-B36-B23*$I$12"

Note that the boundary condition for the slab corner in the north-south direction has changed. The composite structure (beam and slab) remains simply supported because the corner moments are equilibrated internally. However, neither the slab nor the beam have simple supports since they each have boundary moments.

6.4.2.2 Compatibility of deflections at the slab corner

The boundary conditions at the slab corner now include a moment in the north-south direction so that the deflection in cell B34 no longer equals the negative of the deflection in cell B36. If we use the slab moment operator from section 2.2.5 to calculate the corner moment in the east-west direction, we discover that it no longer equals zero (due to Poisson ratio effects of having bending in the north-south direction). We could use the fictitious deflection operator from section 2.3.1 to calculate a new deflection for cell A35 get zero moment in the east-west direction as well:

	A	B	C	D	E	F	G	H
33				NODE DEFLECTIONS (mm)				
34		*-0.622*	*-0.431*	*-0.109*	*0.003*	*-0.109*		
35	*-0.382*	0	**0.3450**	**0.5915**	**0.6803**	0.5915	0.3450	0
36	*0.680*	**0.8392**	**1.0840**	**1.2810**	**1.3548**	1.2810	1.0840	0.8392
37	*1.079*	**1.1733**	**1.3815**	**1.5600**	**1.6282**	1.5600	1.3815	1.1733
38	*0.680*	0.8392	1.0840	1.2810	1.3548	1.2810	1.0840	0.8392
39		0	0.3450	0.5915	0.6803	0.5915	0.3450	0

The operator for beam bending moments developed in section 2.4.1.1 doesn't incorporate Poisson's ratio effects in the transverse direction like the slab moment operator does. It will calculate a beam bending moment at the support because the deflection in cell A35 no longer equals the negative of the deflection in cell C35. The formula for the beam moment at the support was placed in cell B18:

	A	B	C	D	E	F	G	H	I	J
16	Reaction from edge beam:			Smoothing factor: Sb =			17.264		(Sb-1)/Sb =	0.94208
17	Iteration	Mx	R	R	R					
18	current:	42.80	45.29	45.85	46.01					
19	last:		*45.29*	*45.85*	*46.01*					
20		KN-m	KN	KN	KN					

Cell B18 references the slab deflections and beam stiffness as follows:

"=I8/I5*(2*B35-A35-C35)"

The beam end moment was quite large because it is much stiffer than the slab. For obvious reasons a smoothing function is needed for the beam bending formula:

	A	B	C	D	E	F	G	H	I	J
16	Reaction from edge beam:			Smoothing factor: Sb =			17.264		(Sb-1)/Sb =	0.94208
17	Iteration	Mx	R	R	R					
18	current:	2.48	42.81	45.85	46.01					
19	last:	*2.48*	*45.29*	*45.85*	*46.01*					
20		KN-m	KN	KN	KN					

The formula in cell B18 was changed to:

"=I8/I5*(2*B35-A35-C35)**/$G$16+B19*$J$16**"

Cell B19 was set equal to the value in cell B18 as required by the smoothing function highlighted in the above formula.

In order for the composite structure to have a simple support in the east west direction, the beam end moment must be in equilibrium with the slab edge moment at the corner. The beam moment is distributed over h/2 width of slab at the corner so it must be multiplied by 2/h to obtain m_x for the fictitious deflection operator in cell A35:

"=$I$13*B35-$D$11*(B34+B36)-C35+B18*$I$12*2/I5"

6.4.3 Increasing the torsional smoothing factor

When the spreadsheet was iterated after the above modifications had been made, the solution started to diverge. The Author correctly assumed that it had a cause similar to what was encountered in section 5.4.3. A little experimentation showed that the divergence occurred because the beam torsional reactions weren't sufficiently damped by the smoothing factor in cell J22.

The reader should note that the size of the smoothing factor doesn't change the deflections that the solution ultimately converges to. It does, however, affect the number of iterations taken to achieve the solution because it slows down the transfer of forces between the beam and the slab. We must be careful, then, not to arbitrarily pick a huge number such as 1,000,000 because a lot of iterations might be needed to reach the desired accuracy.

Intuitively, the stiffness ratio can be used as a guide when picking a smoothing factor. It can be increased when and if the solution doesn't converge.

We could try doubling the value of the torsional smoothing factor:

	D	E	F	G
21	Smoothing factor: St =			9.22222

The new smoothing factor of 2CG/hD was sufficient to damp out the torsional "over reactions" so the following solution was obtained:

	A	B	C	D	E	F	G	H
33				NODE DEFLECTIONS (mm)				
34		*-0.670*	*-0.372*	*-0.128*	*-0.037*	*-0.128*		
35	*-0.343*	0	**0.3424**	**0.5868**	**0.6748**	0.5868	0.3424	0
36	*0.615*	**0.8032**	**1.0722**	**1.2872**	**1.3675**	1.2872	1.0722	0.8032
37	*1.007*	**1.1313**	**1.3673**	**1.5675**	**1.6438**	1.5675	1.3673	1.1313
38	*0.615*	0.8032	1.0722	1.2872	1.3675	1.2872	1.0722	0.8032
39		0	0.3424	0.5868	0.6748	0.5868	0.3424	0

6.4.4 final solution of the rectangular slab with edge beams having both flexural and torsional stiffness.

After convergence of the deflections, equations from section 2.2.5 were entered to calculate plate bending moments in the north west quadrant:

	A	B	C	D	E	F	G	H	I	J
1	Solution of a rectangular plate with two free edges and two edge beams, simply supported									
2				Mesh size = 4X6						
3										
4	Short span length: LS =			4.000	m	Long span length: LL =			6.000	m
5	Slab thickness: t =			0.300	m		Grid spacing: h =		1.000	m
6	Uniformly distr. load: q =			20.0	kN/m^2	Node load due to q: Pq =			20.000	kN
7		Beam weight: W =		12.0	kN/m	Node load due to W: PW =			12.000	kN
8	Beam moment of inertia: I =			0.040	m^4	Bending stiffness: EI/h^3 =			1160.0	kN/mm
9	B'm torsional constant: C =			0.025	m^4	Torsional stiffness: CG/h^3 =			309.8	kN/mm
10	Young's Modulus: E =			29.0	GPa	Shear Modulus: G =			12.393	GPa
11	Poisson's Ratio: μ =			0.170				D =	67192	kN-m
12								h^2/D =	0.01488	mm/kN
13								2+2μ =	2.340	
14		Iterations/100000 =		0.01911				CG/2h^3 =	154.9	kN/mm
15										
16	Reaction from edge beam:			Smoothing factor: Sb =			17.264	(Sb-1)/Sb =		0.94208
17	Iteration	Mx	R	R	R					
18	current:	0.74	45.06	45.12	45.41					
19	last:	*0.74*	*45.06*	*45.12*	*45.41*					
20		KN-m	KN	KN	KN					
21	Moment from edge beam:			Smoothing factor: St =			9.22222	(St-1)/St =		0.89157
22										
23	m(i) =	-8.968	0.060	2.768	3.311	kN-m/m				
24	m(i-1) =	-8.968	0.060	2.768	3.311	*kN-m/m*				
25										
26		LOADS AT NODES OF THE PRIMARY STRUCTURE (kN)								
27		0	22	22	22	22	22	0		
28		10	20	20	20	20	20	10		
29		10	20	20	20	20	20	10		
30		10	20	20	20	20	20	10		
31		0	22	22	22	22	22	0		
32										
33				NODE DEFLECTIONS (mm)						
34		*-0.670*	*-0.372*	*-0.128*	*-0.037*	*-0.128*				
35	*-0.343*	0	**0.3424**	**0.5868**	**0.6748**	0.5868	0.3424	0		
36	*0.615*	**0.8032**	**1.0722**	**1.2872**	**1.3675**	1.2872	1.0722	0.8032		
37	*1.007*	**1.1313**	**1.3673**	**1.5675**	**1.6438**	1.5675	1.3673	1.1313		
38	*0.615*	0.8032	1.0722	1.2872	1.3675	1.2872	1.0722	0.8032		
39		0	0.3424	0.5868	0.6748	0.5868	0.3424	0		
40										
41	mx	-1.48	6.40	10.68	12.05					
42	kN-m/m	0.00	8.60	13.85	15.54					
43		0.00	9.15	14.73	16.56					
44										
45	my	-8.97	0.06	2.77	3.31					
46	kN-m/m	31.01	29.83	29.76	29.81					
47		42.81	40.07	39.09	38.88					

Figure 22: Solution of rectangular slab with flexural and torsional edge beams

6.4.5 Check of statics

Agreement with static equilibrium is not a proof of the accuracy of our spreadsheet solution. However, any solution which is statically admissible is at least an acceptable lower bound model of the real behavior of the slab-beam system.

At midspan in the east-west direction, the sum of the slab and beam moments must be:

$$\Sigma M = Wl^2/8 = (2*12kN/m + 20kN/m^2*4m)(6m)^2/8 = 468\ kN\text{-}m$$

At midspan in the north-south direction, the slab moments are:

$$\Sigma M = Wl^2/8 = (20kN/m^2*6m)(4m)^2/8 = 240\ kN\text{-}m$$

Finally, the torsion in the edge beam, which equals the edge moment applied to the slab, must sum to zero along the beam.

Formulas entered into the spreadsheet gave the following results which agree with statics:

	A	B	C	D	E	F	G	H	I	J
49					Statics Check					
50	moments in x direction at midspan					moments in y direction at midspan				
51		slab =	59.70	kN-m			slab =	240.00	kN-m	
52		2 beams =	408.30	kN-m						
53		total =	468.00	kN-m	Σedge moments from beam =			0.00	kN-m	
54										

The formula used to sum the slab moments assumed that the distributed moments and torques varied linearly from node to node. Therefore, if we have "n" nodes spaced a distance "h" apart with a moment "m_i" at each node:

$$\Sigma M = h(m_1/2 + m_2 + m_3 + ... + m_{n-1} + m_n/2)$$

Taking advantage of the symmetry, the formulas in cells C51, H51, and H53 were:

```
C51:    "=$I$5*(E43+E41+2*E42)"
H51:    "=$I$5*(B47+E47+2*(C47+D47))"
H53:    "=$I$5*(B45+E45+2*(C45+D45))"
```

The longitudinal bending moment at midspan of the two beams was calculated in cell C52 by using the deflections along the north edge of the slab in the beam bending moment operator from section 2.4.1.1:

```
C52:    "=2*I8*I5*(2*E35-D35-F35)"
```

Interested readers can get the same result by applying the end moment and point loads from cells B18 to E18 on a simply supported, 6m long beam.

this page left blank intentionally

7. DISCUSSION OF RESULTS

7.1 Spreadsheet solutions

The preceding chapters have shown that the spreadsheet/iterative solution to to elastic plate bending is a practical approach to this important class of problems.The examples shown in chapters 4, 5, and 6 can be modelled and solved in under an hour once the user is familiar with the techniques and operators involved.

Plate, beam, and boundary operator equations are graphically visualized by the user because they can be formulated by "pointing" to the nodes on the mesh represented by the spreadsheet cells. These same formulas are easily copied to other cells (or even other spreadsheet models) and modified as necessary to incorporate various beams or other boundary conditions.

7.2 Solution accuracy

Section 4.6 compared the accuracy of the spreadsheet/iterative solution for a simply supported plate against the classical Navier solution. The iterative technique gave satisfactory results for a reasonable number of unknowns and iterations. Two factors control the solution accuracy:

a) The mesh refinement controls the ultimate accuracy which can be achieved if the interdependent system of plate operator equations could be solved exactly. Solutions for the simply supported square plate were within 5% of the Navier bending moments at midspan using meshes as coarse as 4x4.

b) The number of iterations that are performed controls how closely the spreadsheet converges to the "true" solution for the particular mesh refinement. Changes in the deflected shape between subsequent iterations can be used to establish when to stop iterating. The convergence criteria needs to be several decimal places finer than the desired accuracy of the solution, and depends on the rate of convergence (which is related to the number of unknown deflections to be solved.)

7.3 Convergence of iterative finite difference solutions

In the experience of the Author, once the spreadsheet model was correct, it always converged to a solution. Section 5.4.3 discusses saving frequent backup copies of the spreadsheet, particularly before commencing iterations, so that a solution which has diverged to (infinitely) large deflections can easily be recovered.

The author was able to develop a handy Excel® command key macro to reset the displayed cell values zero without changing their underlying operator equations. This was not only useful for recovering from diverged solutions but it gave a consistent basis when comparing the number of iterations required to converge to a specified solution accuracy.

Two situations were encountered where a finite difference formulation failed to converge. The steps taken to correct the model are detailed below:

a) Errors in the operator formulas were a common cause for the spreadsheet model to diverge as the iterations progressed. The errors were usually found by performing hand calculations on the diverged deflections and carefully examining the formulas in any cells which didn't agree with the hand calculations. A common error when superimposing a beam's reactive moments and forces on the primary structure, was to apply them in the wrong direction so that they increase rather than restrain the deflections.

b) When a beam was superimposed onto the slab model by using the reaction between the beam and slab for compatible deflections, the difference in stiffness between the beam and the slab caused the iterations to quickly diverge. Sections 5.4.3 and 6.4.3 discuss a smoothing function which was applied to the model to damp out extreme flexural and torsional reactions from the beam. The required magnitude of the smoothing factor appears to be related to the ratio of beam to slab stiffnesses. For the examples in chapters 5 and 6, making the smoothing factor equal the beam to slab stiffness ratio was sufficient to achieve a solution when a flexural beam was added to the model. When a torsional beam was added to the model in section 6.4.2, the smoothing factor had to be greater than the torsional stiffness ratio. Increasing the magnitude of the smoothing factor does not change the solution which the spreadsheet eventually converges to, but does affect the number of iterations required to get there.

7.4 Other uses for the spreadsheet/iteration technique

The solution method used in this paper may be applied to many other classes of engineering problems, including flow nets for groundwater seepage, slope stability analysis, and heat transfer equations. Any problem which can be formulated to relate the state at one node to the states at surrounding nodes on a regular mesh could potentially be solved iteratively on a spreadsheet.

Within the class of elastic plate bending problems examined in this paper, there are many areas which could be explored further:

- Operators can easily be developed to calculate the relationship between a moment applied at the end of a column and the resulting rotation. Using the compatibility approach developed in this paper, the moment reaction from a column could be applied to the slab (and smoothed if necessary) to incorporate columns with finite stiffness into the model.

- All of the example problems explored in this paper were at least doubly symmetric. Non-symmetric problems can easily be formulated but will have more deflection unknowns to be solved. For instance, "simple" column supports at the interior of a slab can be placed at random locations by setting the deflection of those nodes to zero. It also might be interesting to calculate the deflected shape and stresses in the structure analyzed in chapter 6 if one of the corners is given a differential settlement by specifying a non zero deflection.

- Three dimensional rectangular storage containers with hydrostatic loading can easily be modelled since the deflections adjacent to the corners on each face of the structure are related to each other by compatibility of slopes and moment equilibrium.

- Cantilever slabs can be analyzed using the free edge and corner operators developed in section 2.3.4.

7.5 The effect of future improvements in microcomputers

The only practical limitation on the use of this method is the number of iterations required to converge to a satisfactory solution. Chapter 4 showed that the MacIntosh® microcomputer used by the Author could solve about 30 unknown deflections to an accuracy of 1% in under an hour. Newer models of all makes of computers are starting to use faster processors and co-processors to achieve hundredfold increases in computing speed. The next generation of microcomputers might practically solve several hundred unknowns in the less than an hour. As the technology improves, standard spreadsheet programs will be able to analyze many complex structures which can only be solved using dedicated finite element programs at the present time.

this page left blank intentionally

REFERENCES

1 Ang, A. H. -S., and Newmark, M. N., " A Numerical Procedure for the Analysis of Continuous Plates,"Proceedings, 2nd ASCE Conf. on electronic Computation, September 1960, p.379.

2 Salvadori, M. G., and Baron, M. L., "Numerical Methods in Engineering," Prentice-Hall, inc., Englewood Cliffs, N.J., 1952

3 Wood, R. H., "Plastic and Elastic Design of Slabs and Plates," Thames and Hudson, London, 1961

4 Prescott, W. S., Ang, A., and Siess, C. P., "Analysis of Clamped Square Plates Containing Openings with Stiffened Edges", University of Illinois Civil Engineering Studies, Structural Research Series No. 229, November 1961.

5 Simmonds, S. H., and Siess, C. P., "Effects of Column Stiffness on the Moments in Two Way Floor Slabs," University of Illinois Civil Engineering Studies, Structural Research Series No. 253, July 1962.

6 Timoshenko, S. P., and Woinowsky-Krieger, S., "Theory of Plates and Shells," Second Edition, McGraw-Hill Book Co., New York, 1959

7 Ang, Alfredo H. -S., and Prescott, Wallace, "Equations for Plate-Beam Systems in Transverse Bending," A.S.C.E. Proceedings, Engineering Mechanics Division Journal, December 1961

8 Newmark, N. M., "Numerical methods of Analysis of Bars, Plates, and Elastic Bodies," An article from "Numerical Methods of Analysis in Engineering," Edited by L. E. Grinter, MacMillan Co., New York, 1949.

9 Jaeger, G., "Elementary Theory of Elastic Plates," Pergamon Press Ltd., Oxford, 1964.

10 Irons, B., and Ahmad, S., "Techniques of Finite Elements," John Wiley & Sons Canada Ltd., Rexdale, 1980.

RECENT STRUCTURAL ENGINEERING REPORTS

Department of Civil Engineering

University of Alberta

118. *Effective Lengths of Laterally Unsupported Steel Beams* by C.D. Schmitke and D.J.L. Kennedy, October 1984.

119. *Flexural and Shear Behaviour of Large Diameter Steel Tubes* by R.W. Bailey and G.L. Kulak, November 1984.

120. *Concrete Masonry Prism Response due to Loads Parallel and Perpendicular to Bed Joints* by R. Lee, J. Longworth and J. Warwaruk.

121. *Standardized Flexible End Plate Connections for Steel Beams* by G.J. Kriviak and D.J.L. Kennedy, December 1984.

122. *The Effects of Restrained Shrinkage on Concrete Slabs* by K.S.S. Tam and A. Scanlon, December 1984.

123. *Prestressed Concrete Beams with Large Rectangular Web Openings* by T. do M.J. Alves and A. Scanlon, December 1984.

124. *Tests on Eccentrically Loaded Fillet Welds* by G.L. Kulak and P.A. Timler, December 1984.

125. *Analysis of Field Measured Deflections Scotia Place Office Tower* by A. Scanlon and E. Ho, December 1984.

126. *Ultimate Behaviour of Continuous Deep Reinforced Concrete Beams* by D.R. Ricketts and J.G. MacGregor, January 1985.

127. *The Interaction of Masonry Veneer and Steel Studs in Curtain Wall Construction* by W.M. McGinley, J. Warwaruk, J. Longworth and M. Hatzinikolas, May 1985.

128. *Evaluation of Existing Bridge Structure by Nondestructive Test Methods* by L. Mikhailovsky and A. Scanlon, May 1985.

129. *Finite Element Modelling of Buried Structures* by D.K. Playdon and S.H. Simmonds, October 1985.

130. *Behaviour and Ultimate Strength of Transversely Loaded Continuous Steel Plates* by K.P. Ratzlaff and D.J.L. Kennedy, November 1985.

131. *Inelastic Lateral Buckling of Steel Beam-Columns* by P.E. Cuk, M.A. Bradford and N.S. Trahair, December 1985.

132. *Design Strengths of Steel Beam-Columns* by N.S. Trahair, December 1985.

133. *Behaviour of Fillet Welds as a Function of the Angle of Loading* by G.S. Miazga and D.J.L. Kennedy, March 1986.

134. *Inelastic Seismic Response of Precast Concrete Large Panel Coupled Shear Wall Systems* by M.R. Kianoush and A. Scanlon, March 1986.

135. *Finite Element Prediction of Bin Loads* by A.H. Askari and A.E. Elwi, June 1986.

136. *Shear Behavior of Large Diameter Fabricated Steel Cylinders* by J. Mok and A.E. Elwi, June 1986.

137. *Local Buckling Rules for Structural Steel Members* by S. Bild and G.L. Kulak, May 1986.

138. *Finite Element Prediction of Reinforced Concrete Behavior* by S. Balakrishnan and D.W. Murray, July 1986.

139. *Behavior and Strength of Masonry Wall/Slab Joints* by T.M. Olatunji and J. Warwaruk, July 1986.

140. *Bayesian Analysis of In-Situ Test Data for Estimating the Compressive Strength of Concrete in Existing Structures* by G.J. Kriviak and A. Scanlon, July 1986.

141. *Shear-Moment Transfer in Slab-Column Connections* by S.D.B. Alexander and S.H. Simmonds, July 1986.

142. *Minimum Thickness Requirements for Deflection Control of Two-Way Slab Systems* by D.P. Thompson and A. Scanlon, November 1986.

143. *Shrinkage and Flexural Tests of Two Full-Scale Composite Trusses* by A. Brattland and D.J.L. Kennedy, December 1986.

144. *Combined Flexure and Torsion of I-Shaped Steel Beams* by R.G. Driver and D.J.L. Kennedy, March 1987.

145. *Cyclic and Static Behaviour of Thin Panel Steel Plate Shear Walls* by E.W. Tromposch and G.L. Kulak, April 1987.

146. *Postbuckling Behavior of Thin Steel Cylinders Under Transverse Shear* by V.G. Roman and A.E. Elwi, May 1987.

147. *Incipient Flow in Silos - A Numerical Approach* by R.A. Link and A.E. Elwi, May 1987.

148. *Design of Web-Flange Beam or Girder Splices* by D. Green and G.L. Kulak, May 1987.

149. *Spreadsheet Solution of Elastic Plate Bending Problems* by G.E. Small and S.H. Simmonds, July 1987.