25

PRESLAB - MICRO-COMPUTER ANALYSIS AND DESIGN

OF

PRESTRESSED CONCRETE SLABS

by

André Johan du Toit

Submitted to the University of Cape Town in partial fulfilment of the requirements for the degree of Master of Science in Engineering.

Department of Civil Engineering
University of Cape Town

June 1988

DECLARATION OF CANDIDATE

I hereby declare that this thesis is my own work and that it has not been submitted for a degree at any other University.

Signed by candidate

.....

A J du Toit

June 1988

ACKNOWLEDGEMENTS

I would like to extend my sincerest appreciation to the following people :

Professor W S Doyle, under whose supervision this thesis was conducted.

Cheryl Wright for typing this thesis.

Oscar Shouw for drawing the figures.

ABSTRACT

A micro-computer based package for the analysis and design of prestressed flat slabs is presented.

The constant strain triangle and the discreet Kirchhoff plate bending triangle are combined to provide an efficient "shell" element. These triangles are used for the finite element analysis of prestressed flat slabs.

An efficient out-of-core solver for sets of linear simultaneous equations is presented. This solver was developed especially for micro-computers.

Subroutines for the design of prestressed flat slabs include the principal stresses in the top and bottom fibres of the plate, Wood/Armer [39,40] moments and untensioned steel areas calculated according to Clark's [47] recommendations.

Extensive pre- and post-processing facilities are presented.

Several plotting routines were developed to aid the user in his understanding of the behaviour of the structure under load and prestressing.

CONTENTS

NOMENCLATURE

Upper Case Characters

A	plan area of an element
A_s	cross-sectional area of a prestressing cable
B	strain matrix
D	elasticity matrix or diagonal matrix
D_b	elasticity matrix for bending
D_s	elasticity matrix for shear
$[DL]$	transformation matrix i.e. $[B^T][D][B] = [x][DL][x]$
E	Young's modulus
E_s	Young's modulus of a prestressing cable
F	load vector
F_c	compressive force in concrete per unit length
F_e	equivalent nodal load vector
F_x	prestressing force in a cable at distance x
F_o	jacking force in a cable
G_x, G_y	matrix of shape functions independent of element coordinates
H	matrix of shape functions
K	global stiffness matrix
K_e	element stiffness matrix
M	bending moment vector
M_x, M_y, M_{xy}	bending moment triad
M_x, M_α	transformed moments in the x and α directions
N_x, N_y, N_{xy}	in-plane force triad
N_x, N_α	transformed in-plane forces in the x and α directions
N_i	interpolation functions
Q	shear stress vector
R	constant matrix

U	upper triangular matrix
U_b	bending strain energy
U_s	shear strain energy
U^e	vector of in-plane nodal displacements in the X direction
V^e	vector of in-plane nodal displacements in the Y direction
X,Y,Z	transformed interpolation matrices

Lower Case Characters

dL	wedge set of prestressing system
k	"wobble" coefficient for prestressing
l	wedge set length
p	applied body forces per unit volume
q	applied surface tractions
u	displacements in the X direction
v	displacements in the Y direction
w	transverse displacements

Greek Characters

α	shear correction factor
$[\alpha]$	transformation matrix i.e. $[B^T][D][B] = [x][DL][x]$
β_X, β_Y	rotations of the normal to the undeformed midsurface
δ	displacement vector
δ^e	nodal displacement vector
ε	strain vector
ε_b	bending strain
ε_s	shear strain
γ	transverse shear strain vector
κ	curvature vector

Greek Characters (continued)

ξ,η	area or curvilinear coordinates
π	total potential energy
π_e	potential energy of an element
σ	stress vector
σ_b	bending stress vector
σ_s	shear stress vector
$\sigma_x,\sigma_y,\tau_{xy}$	stress triad at a point in a plate
σ_I,σ_{II}	principal stresses
ν	Poisson's ratio
θ	angular rotation of a prestressing cable
μ	curvature friction coefficient of a prestressing cable
Φ	vector of area coordinates

CHAPTER 1

INTRODUCTION

'Computation assists engineering judgement but must not replace it' [2]

Prestressed flat slabs are becoming ever more popular due to their relative economy over ordinary reinforced concrete. The design of these elements have, to date, rested on simplified frame analysis and/or design rules [47].

Flat slabs are, in fact, nearly always statically indeterminate structures with a complex response to loading conditions. Cut-outs, stiff walls, columns and prestressing can complicate the behaviour of the slab beyond the capabilities of simple approximations. When this happens, alternative means of analysis must be sought. The Finite Element Method (FEM) is ideally suited to this type of analysis.

The FEM has always been recognized as the premier mathematical tool for structural analysis. Up to a few years ago, mainframe computers completely dominated the computational basis of this method.

Micro-computers, due to their very competitive price and phenomenal increase in power, have started somewhat of a revolution in structural engineering. Unfortunately, micro-computer based software support for the FEM lagged behind and the revolution seemed to be doomed in its infancy.

"PRESLAB" was developed as a unified approach to the problem of prestressed flat slab analysis and design on micro-computers. The aim of the development was to provide the design engineer with sophisticated tools which will :

- Aid in the analysis of complex flat slabs under any number of load cases without requiring an in-depth understanding of the FEM.

- Present data and results visually, in the form of graphic plots with which design engineers are familiar. This gives the design engineer greater insight into the behaviour and response of the slab.

- Provide facilities to calculate the loading due to prestressing, principal stresses and untensioned steel requirements. These facilities remove much of the tedious calculations required by the design.

The program was built within the limitations of micro-computers while utilizing most of the features that make these small machines remarkable.

The "PRESLAB" suite of programs comprises the following four sections :

a) Super element section

Rapid generation of a finite element mesh through the use of super elements.

b) System section

Set up and triangulate the structure stiffness matrix. The analysis incorporates a recently developed three-noded triangular plate bending element based on the discreet Kirchhoff theory. A constant strain triangle is superimposed on the bending element. With this formulation, both bending and in-plane effects are modelled on the same element. The element is described in Chapter 2.

An out-of-core profiled solver was developed for the solution of a system of linear algebraic equations. The solver is based on the Cholesky method of decomposition and was developed especially for micro-computers. A complete description of the solver follows in Chapter 6.

c) Loadcase section

Define loadcases, set up loadvectors and calculate deflections, stresses and reactions for these loadcases.

d) Post-processing section

Loadcases can be combined, prestressing profiles and their effects calculated, principal stresses calculated, untensioned steel designed and results printed. Deflected shapes and vector plots of results

can also be viewed. The design of untensioned reinforcement is based on formula as proposed by Nielsen [39], Wood [40], Armer [41] and Clark [48].

Pre-processing consists of powerful data editors and graphic presentation of the finite element mesh.

Verification examples for the element behaviour are included. A design example illustrates the various features and presents the design procedure envisaged with this development.

CHAPTER 2

FINITE ELEMENT THEORY

2.1 Basic principles of the finite element method

In the finite element displacement method the governing differential equations for a continuous system is approximated by a set of algebraic equations relating the displacements of a finite number of points on the system.

The physical domain of a problem is divided up into sub-domains called 'finite elements'. These elements contain points on the boundary or in the interior of the element called 'nodes'. The displacements at any point on the element is described in terms of the displacements at these nodes.

Equations that govern the equilibrium of displacements and applied load conditions of an element can be derived by minimizing the total potential energy of the system. The principle of stationary potential energy states :

> 'for all geometrically compatible deformation states of a structural system that also satisfy the deflection boundary conditions, those that also satisfy the force equilibrium requirements give stationary and minimum values to the total potential energy'. [5]

The total potential energy of an element can be expressed in general terms as :

$$\pi = \tfrac{1}{2} \int_V \sigma^T \varepsilon \, dV - \int_V \delta^T p \, dV - \int_S \delta^T q \, dS \qquad (2.1)$$

where :

- σ is the stress vector,
- ε is the strain vector,
- δ is the displacements at any point on the structure,
- p is the applied body forces per unit volume and
- q is the applied surface tractions.

$\int_V$ and $\int_S$ represent volume and surface integrals respectively.

The first term on the right hand side of equation (2.1) is referred to as the strain energy term, U. The second and third terms accounts for the work contributions from the body forces and surface tractions respectively.

Displacements at any point in the element are related to the displacements of the nodal points by :

$$\delta = N \delta^e \tag{2.2}$$

where :

- N represents the interpolation functions and
- δ^e is the vector of nodal displacements of the element.

The strain matrix (B) relates the strains anywhere within the element to the nodal displacements by :

$$\varepsilon = B \delta^e \tag{2.3}$$

Stresses in the elements are related to the strains by means of the elasticity matrix, (D) i.e. :

$$\sigma = D\varepsilon \tag{2.4}$$

Using (2.1-2.4), the total potential energy of an element can be expressed as :

$$\pi_e = \tfrac{1}{2} \int_{V_e} \delta^e B^T DB \, \delta^e \, dV - \int_{V_e} \delta^{eT} N^T p dV - \int_{S_e} \delta^{eT} N^T q dS \tag{2.5}$$

Since the nodal displacements are the primary unknowns, the nodal degrees of freedom must assume such values that the total potential energy is stationary. Differentiating equation (2.5) with respect to the nodal displacements and equating to zero, gives :

$$\frac{\partial \pi_e}{\partial \delta^e} = \int_{V_e} B^T D B \, \delta^e \, dV - \int_{V_e} N^T p dV - \int_{S_e} N^T q dS \qquad (2.6)$$

or,

$$= K^e \delta^e - F_e \qquad (2.7)$$

where :

- F_e is the vector of equivalent nodal forces and
- K the element stiffness matrix.

Summing equation (2.6) over all the elements in the discretization, results in a system of simultaneous linear equations. This system, when solved, yields the nodal displacements. Stresses, strains and displacements anywhere within the element can be calculated from these nodal displacements using (2.2-2.4).

2.2 Choice of element

Selection of a suitable element for the current development was based on the following criteria :

a) The element must always yield a stable solution that converges to the correct solution under all conditions. It is impractical to apply 'rules' that tell under what boundary conditions and for which structural shapes an element can be employed.

It is also necessary that the finite element mesh should converge to the correct solution on successive refinement of the mesh. The correct solution should preferably be approached monotonically. That is, the element must be complete and preferably compatible.

b) The element must have at least two rotational and three translational degrees of freedom per node in order to include both the axial and bending components of a prestressing system on one element.

Traditionally this was achieved by doing separate finite element analyses for the in-plane effects and the plate bending effects or by utilising general shell elements in the formulation.

Using two finite element models for one analysis is rather cumbersome and time-consuming, especially on micro-computers. General shell elements, on the other hand, couple the in-plane and bending effects in the element stiffness matrix. These effects are uncoupled in flat plates which means that a lot of unnecessary computational effort is spent on calculating essentially zero terms in the stiffness matrix.

c) The element must be cost efficient. On micro-computers the cost efficiency of an element formulation is determined by :

- the computational effort required to calculate the stiffness matrix.

- the number of lines of code required to formulate the matrix and

- the computational effort required to retrieve the required stresses once the displacement vector has been calculated.

d) The element must be able to model arbitrary plate geometry, general supports and cut outs. In this regard, isoparametric and triangular elements are particularly effective.

e) The element should be able to accommodate beam stiffeners. Although not included in the current development, it is important to select degrees of freedom for the plate elements that are compatible with those of beam stiffeners.

In his very definitive paper on plate bending elements, Batoz [6] mentions the following three approaches to the formulation of plate and shell elements :

a) A particular shell theory is used and discretized.

b) Three-dimensional curvilinear continuum equations are used and discretized (isoparametric elements).

c) Plate bending and membrane element stiffnesses are superimposed and assembled in a global coordinate system.

Considering the practical requirements listed above, approach 3 was deemed the most suited to the analysis of prestressed flat slabs.

2.3 Plate bending element

2.3.1 Introduction

Batoz [6] concludes that the discrete Kirchhoff theory (DKT) and hybrid stress model (HSM) elements are the most efficient, cost effective and reliable 9 degree of freedom triangular elements for bending analysis of thin plates. In a further paper Batoz [7] shows that the HSM element deteriorates with an increase in the element aspect ratio while the DKT element remains relatively well behaved. He also presents an explicit formulation of the DKT element stiffness matrix in local coordinate system of which he says :

> "this formulation should be attrative for finite element programs on desk top and micro-computers" [6].

In their paper, Jeyachandrabose and Kirkhope [9] present an explicit formulation of the DKT element in global coordinates. This formulation avoids the matrix triple product required for coordinate transformation and makes the element even more attractive for micro-computer applications.

All three formulations of the DKT element were programmed and tested for an arbitrary homogeneous isotropic material. It is surprising to note that the assembly of the implicit formulation in global coordinates was the fastest, followed by the explicit formulation in local coordinates with the slowest being the explicit formulation in global coordinates. The explicit formulation in global coordinates required by far the least number of lines to program.

As run times on micro-computers are very much hard- and soft-ware dependent, it was decided to use the explicit formulation in global coordinates in view of the tremendous saving in the number of programming lines.

2.3.2 Formulation of the discrete Kirchhoff theory element

The formulation of the DKT element is well documented [6-9] and will only be summarized here. The evolution of the explicit formulation of the stiffness matrix in global coordinates from the original implicit form presented by Batoz [6] will also be reviewed.

2.3.2.1 Theory of plates with small displacements including transverse shear

The plate theory of Reissner or Mindlin is based on the following assumptions :

a) The deflections of the plate are small.

b) Normals to the midsurface before deformation remain straight but not necessarily normal to the midsurface after deformation (Kirchhoff hypothesis).

c) Stresses normal to the midsurface are negligible irrespective of the loading (plane stress assumption).

If the undeformed midsurface lies in the X-Y plane, the displacement components of a point with coordinates x,y,z are given by :

$$u = z\,\beta_x\,(x,y)$$

$$v = z\,\beta_y\,(x,y) \qquad (2.8)$$

$$w = w\,(x,y)$$

where :

- w is the transverse displacement and
- β_x and β_y are the rotations of the normal to the underformed midsurface in the X-Z and Y-Z planes respectively (Figure 2.1).

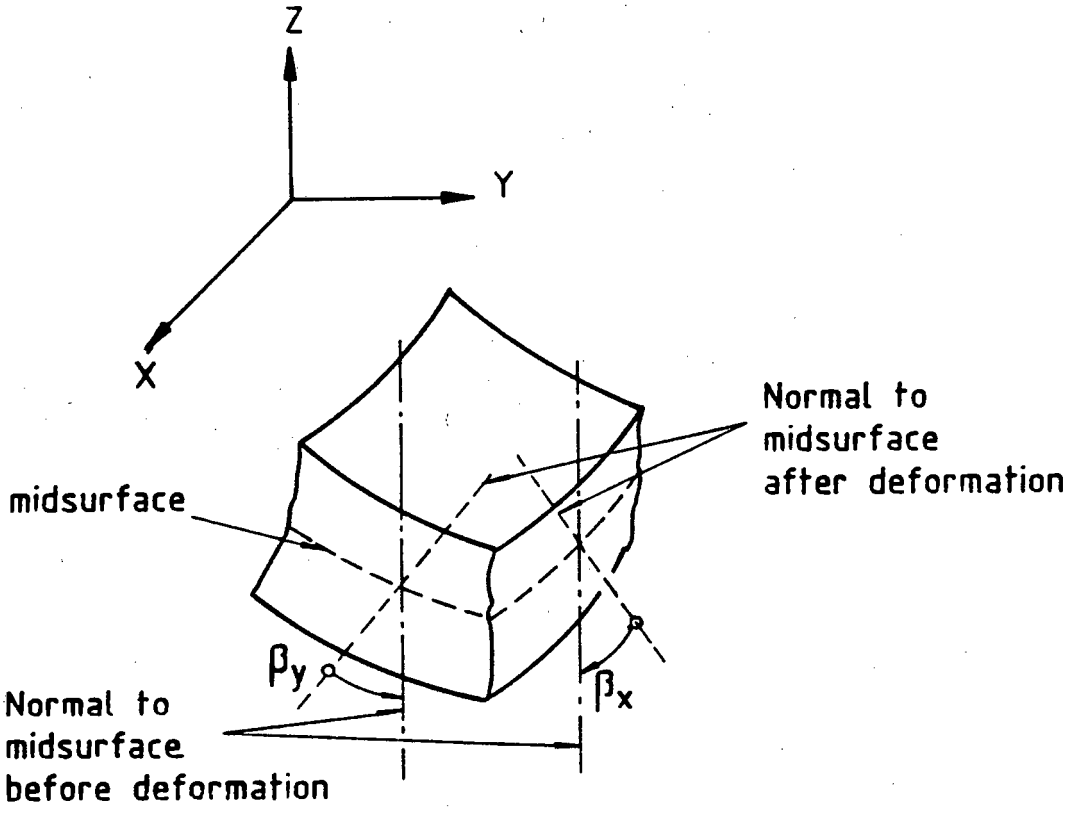


Figure 2.1 : Positive directions of β_x and β_y .

The bending strain varies linearly through the thickness of the plate and is :

$$\varepsilon_b = z\kappa \tag{2.9}$$

and κ is the curvature vector, i.e. :

$$\kappa = \begin{bmatrix} \beta_{x,x} \\ \beta_{y,y} \\ \beta_{x,y} + \beta_{y,x} \end{bmatrix} \tag{2.10}$$

Transverse shear strains are constant through the thickness of the plate and are :

$$\gamma = \begin{bmatrix} w_{,x} + \beta_x \\ w_{,y} + \beta_y \end{bmatrix} \tag{2.11}$$

With the assumptions of plane stress and no coupling between the bending and shear deformations, the bending stresses are :

$$\sigma_b = zD\kappa \tag{2.12}$$

and the shear stresses are :

$$\sigma_s = E\gamma \tag{2.13}$$

The strain energy associated with the kinematics in (2.8-2.11) and material description in (2.12) and (2.13) is, for bending :

$$U_b = \tfrac{1}{2} \int_A \kappa^T D_b \kappa \, dxdy \tag{2.14}$$

and for shear :

$$U_s = \tfrac{1}{2} \int_A \gamma^T D_s \gamma \, dxdy \tag{2.15}$$

For an isotropic homogeneous plate of constant thickness (h) :

$$D_b = \int_{\frac{-h}{2}}^{\frac{h}{2}} D(z) \, z^2 \, dz$$

$$= \begin{bmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \end{bmatrix} \tag{2.16}$$

$$= \frac{Eh^3}{12(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}$$

and

$$D_s = \alpha \int_{\frac{-h}{2}}^{\frac{h}{2}} E(z) \, dz \, \alpha$$

$$= \frac{\alpha E h}{2(1+\nu)} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{2.17}$$

where :

- E is Young's modulus,
- ν is Poisson's ratio and
- α is the shear correction factor.

Bending moments, M and shear stresses, Q are obtained by integration of the stresses through the thickness of the plate :

$$M = \begin{bmatrix} M_x \\ M_y \\ M_{xy} \end{bmatrix} = \int_{-\frac{h}{2}}^{\frac{h}{2}} \sigma_b \, z dz = D_b \kappa \qquad (2.18)$$

$$Q = \begin{bmatrix} Q_x \\ Q_y \end{bmatrix} = \alpha \int_{-\frac{h}{2}}^{\frac{h}{2}} \sigma_s \, dz\alpha = D_s \gamma \qquad (2.19)$$

2.3.2.2 Implicit formulation of the DKT element stiffness matrix in global coordinates

For thin plates the shear strains and hence the shear strain energy, is negligible compared to the bending strain energy. The strain energy can, therefore, be approximated by :

$$U = U_b = \tfrac{1}{2} \int_A \kappa^T D_b \kappa \, dxdy \qquad (2.20)$$

Consider the six noded triangular element in Figure 2.2.

If β_x and β_y vary quadratically over the element, then :

$$\beta_x = \sum_{i=1}^{6} N_i \beta_{xi} \qquad (2.21)$$

$$\beta_y = \sum_{i=1}^{6} N_i \beta_{yi} \qquad (2.22)$$

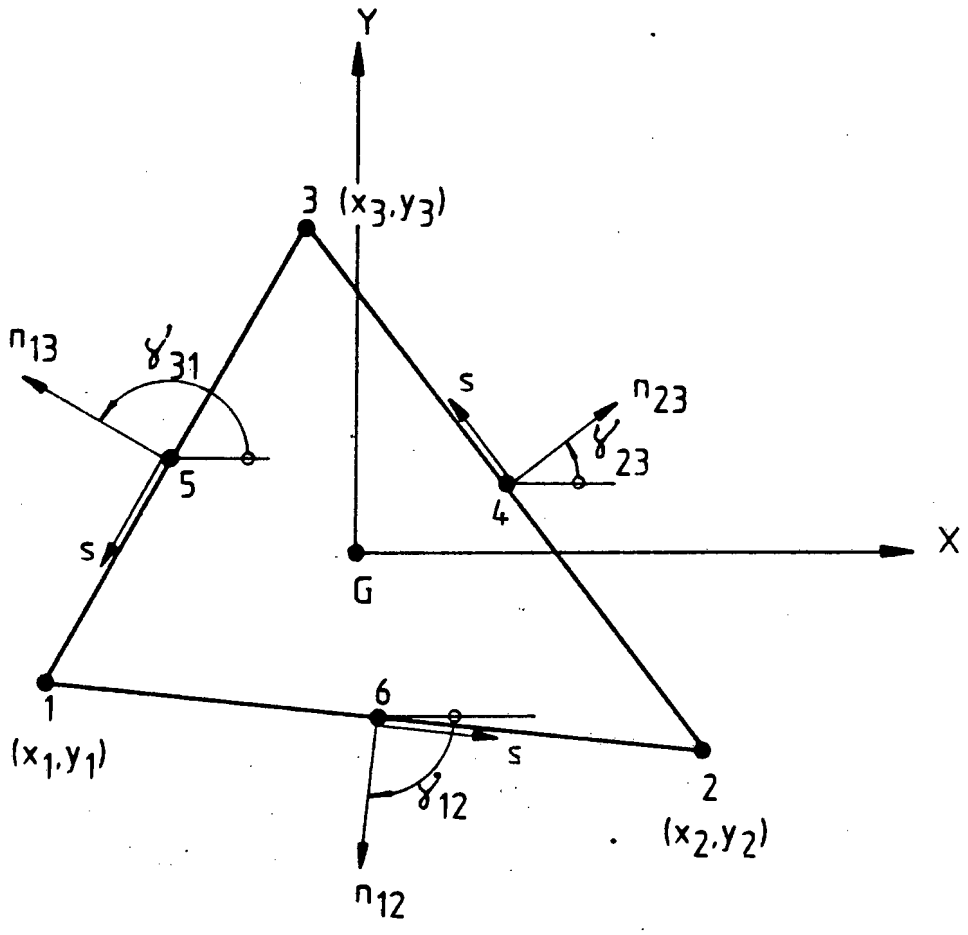


Figure 2.2 : Geometry of the triangular element

where :

- β_{xi} and β_{yi} are the nodal values and
- N_i (ξ,η) are the shape functions in area coordinates, i.e. :

$$
\begin{aligned}
N_1 &= 2\,(1-\xi-\eta)\,(0.5-\xi-\eta) \\
N_2 &= \xi\,(2\xi-1) \\
N_3 &= \eta\,(2\eta-1) \\
N_4 &= 4\xi\eta \\
N_5 &= 4\eta\,(1-\xi-\eta) \\
N_6 &= 4\xi\,(1-\xi-\eta)
\end{aligned}
\qquad (2.23)
$$

Imposing the Kirchhoff hypothesis at the corner and midside nodes, we have :

$$\gamma = \begin{bmatrix} \beta_x + w_{,x} \\ \beta_y + w_{,y} \end{bmatrix} = 0 \quad \text{(at nodes 1,2 and 3)} \tag{2.24}$$

and at the midside nodes :

$$\beta_{sk} + w_{,sk} = 0 \quad (k = 4,5,6) \tag{2.25}$$

Assuming that the variation of w along any side is cubic, the displacement at the midside nodes are :

$$w_{,sk} = -\frac{3}{2}\frac{w_i}{\ell_{ij}} - \frac{1}{4} w_{,si} + \frac{3}{2}\frac{w_i}{\ell_{ij}} - \frac{1}{4} w_{,si} \tag{2.26}$$

w_k is located midway on side ij and ℓ_{ij} is the length of side ij .

The transverse shear strain at three points along each side of the element is zero. Since $w_{,s}$ and β_s vary quadratically along the side, $w_{,s}$ matches β_s and the Kirchhoff hypothesis ($\gamma = \beta_s + w_{,s} = 0$) is satisfied along the entire boundary of the element.

Imposing a linear variation of β_n on the sides, we have :

$$\beta_{nk} = \tfrac{1}{2} (\beta_{ni} + \beta_{nj}) \tag{2.27}$$

where k (= 4,5,6) denotes the midside nodes of sides 23, 31 and 12 respectively.

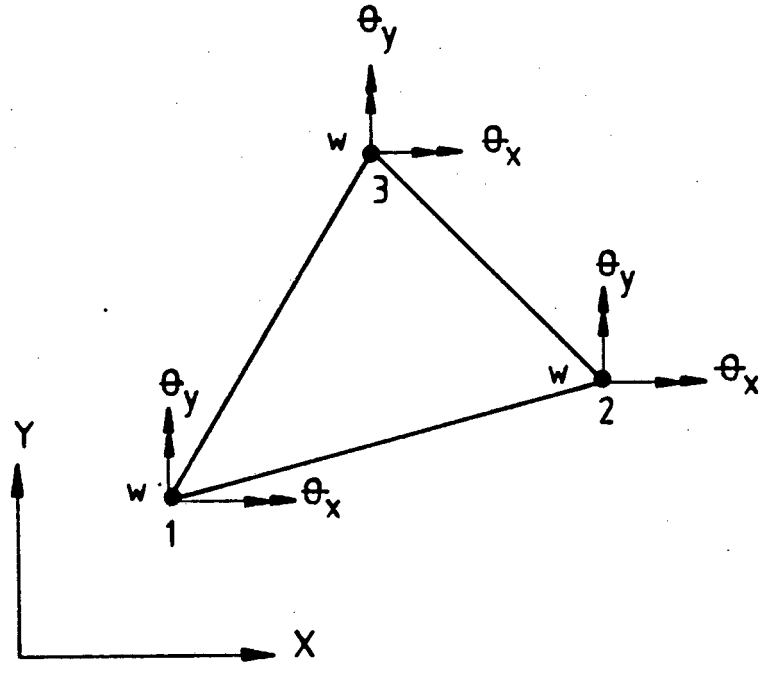


Figure 2.3 : Degrees of freedom of the DKT element (global coordinates)

Relating the global degrees of freedom to those on each side of the element, we have :

$$\begin{bmatrix} \beta_x \\ \beta_y \end{bmatrix} = \begin{bmatrix} c & -s \\ s & c \end{bmatrix} \begin{bmatrix} \beta_n \\ \beta_s \end{bmatrix} \tag{2.28}$$

where :

$$c = \cos \gamma_{ij} \quad \text{and} \quad s = \sin \gamma_{ij}$$

For the nodal degrees of freedom in Figure 2.3,

$$\delta^e = [w_1 , \theta_{x1} , \theta_{y1} , w_2 , \theta_{x2} , \theta_{y2} , w_3 , \theta_{x3} , \theta_{y3}] \tag{2.29}$$

and using equations (2.21)-(2.28), we can write :

$$\beta_x = H_x^T(\xi,\eta)\,\delta^e \qquad \beta_y = H_y^T(\xi,\eta)\,\delta^e \tag{2.30}$$

H_x and H_y are derived from the shape functions and appear in reference [6].

From equations (2.10) and (2.30) it follows that :

$$\kappa = B\delta^e \tag{2.31}$$

B represents the strain-displacement transformation matrix :

$$B(\xi,\eta) = \frac{1}{2A}\begin{bmatrix} y_{31}\,H^T_{x,\xi} + y_{12}\,H^T_{x,\eta} \\ -x_{31}\,H^T_{y,\xi} - x_{12}\,H^T_{y,\eta} \\ -x_{31}H^T_{x,\xi} - x_{12}H^T_{x,\eta} + y_{31}H^T_{y,\xi} + y_{12}H^T_{y,\eta} \end{bmatrix} \tag{2.32}$$

and $2A = x_{31}\,y_{12} - x_{12}\,y_{31}$

Derivatives of H_x and H_y appear explicitly in Batoz's paper [6].

The stiffness matrix of the DKT element then becomes :

$$K = 2A\int_0^1\int_0^{1-\eta} B^T D_b\, B\, d\xi\, d\eta \tag{2.33}$$

For an element with constant thickness and isotropic homogeneous material properties, the stiffness matrix can be assembled by applying quadrature formulae at three numerical integration points on the element.

Bending moments M at any point on the element follow from the nodal displacements and are :

$$M(x,y) = D_b \, B(\xi,\eta) \, \delta^e \tag{2.34}$$

where for the coordinate system in Figure 2.3 :

$$\begin{aligned} x &= x_1 + \xi \, x_{21} + \eta \, x_{31} \\ y &= y_1 + \xi \, y_{21} + \eta \, y_{31} \end{aligned} \tag{2.35}$$

2.3.2.3 Explicit formulation of the DKT element stiffness matrix in local coordinates

This formulation of the element is based on a procedure proposed by Joseph and Rao [8]. The matrix triple product $[B]^T [D][B]$ is mapped to the form $[\alpha][DL][\alpha]$ where the matrix [DL] is independent of the geometry of the element.

Consider the local coordinate system in Figure 2.4 with the local x axis defined by side 12.

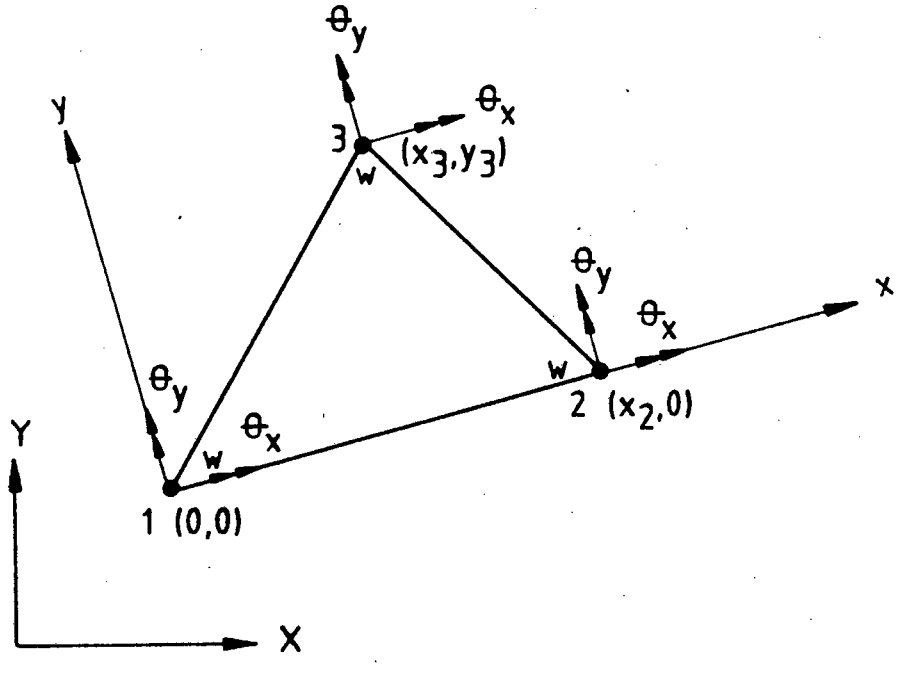


Figure 2.4 : Degrees of freedom of the DKT element (local coordinates)

Matrix α is derived from equation (2.36) by considering each term of B i.e. :

$$B_{ij}(\xi,\eta) = \frac{1}{2A}\left[(1-\xi-\eta)\,\alpha_{kj} + \xi\,\alpha_{k+1,j} + \eta\,\alpha_{k+2,j}\right] \tag{2.41}$$

where $k = 3(i-1) + 1$.

Introducing equation (2.38) into equation (2.33) with $d\,x\,dy = 2A\,d\xi\,d\eta$, an explicit form of the stiffness matrix in local coordinates is obtained :

$$K = \frac{1}{2A}\,\alpha^T\,DL\,\alpha \tag{2.42}$$

with

$$DL = \int_0^1 \int_0^{1-\xi} L^T\,D_b L\,d\xi\,d\eta \tag{2.43}$$

For a homogeneous isotropic material DL is :

$$DL = \frac{1}{24}\begin{bmatrix} D_{11}R & D_{12}R & D_{13}R \\ D_{21}R & D_{22}R & D_{23}R \\ D_{31}R & D_{32}R & D_{33}R \end{bmatrix} \tag{2.44}$$

with :

$$R = \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} \tag{2.45}$$

and the terms of the material matrix D as defined by equation (2.16).

Matrix DL in equation (2.44) is determined only by the material properties and is constant for elements of the same thickness. The matrix α appears in Table 1 of reference [7] and the element stiffness matrix follows from (2.42).

Batoz [7] shows how the matrix product [α][DL] can be manipulated to further reduce the number of algebraic operations in the calculation of the stiffness matrix. This manipulation destroys the independence of the matrix [DL] from the element geometry and for an isotropic homogenious plate of constant thickness, is less efficient.

Bending moments (M) at any point on the element can be calculated from the nodal displacements in local coordinates by :

$$M = \frac{1}{2A} D_b \, L \, \alpha \, \delta^e \tag{2.46}$$

2.3.2.4 Explicit formulation of the DKT element stiffness matrix in global coordinates

The explicit formulation of the DKT element finally used in the program was derived by Jeyachandrabose et al [9].

Consider the bending strain energy of the element U (equation (2.20)), the curvature vector κ (equation (2.10)) and the rotations of the normal to the undeformed midsurface β_x and β_y i.e. :

$$\begin{aligned} \beta_x &= H_x^T (\xi,\eta) \, \delta^e \\ \beta_y &= H_y^T (\xi,\eta) \, \delta^e \end{aligned} \tag{2.47}$$

Explicit formulations of H_x and H_y appear in reference [7] and can be rearranged to take the form :

$$H_x = [1 \; , \; \xi \; , \; \eta \; , \; \xi^2 \; , \; \xi\eta \; , \; \eta^2] \; G_x$$
$$H_y = [1 \; , \; \xi \; , \; \eta \; , \; \xi^2 \; , \; \xi\eta \; , \; \eta^2] \; G_y \qquad (2.48)$$

where G_x and G_y are of the order 6x9 and are independent of ξ and η .

In order to establish the terms in the curvature vector (equation (2.10)), the following transformation rules are applied :

$$\frac{dH}{dx}(\xi,\eta) = \frac{1}{2A}\left[b_2 \frac{dH}{d\xi} + b_3 \frac{dH}{d\eta}\right]$$
$$\frac{dH}{dy}(\xi,\eta) = \frac{1}{2A}\left[c_2 \frac{dH}{d\xi} + c_3 \frac{dH}{d\eta}\right] \qquad (2.49)$$

where $b_i = (y_j - y_k)$ and $c_i = (x_k - x_j)$, with (i,j,k) an even permutation of (1,2,3) i.e. :

$i = 2 \quad j = 3 \quad k = 1$
$i = 3 \quad j = 1 \quad k = 2$

From equations (2.47) and (2.48) it follows that :

$$\beta_{x,x} = \frac{1}{2A}(1,\xi,\eta) \begin{bmatrix} b_2\langle G_x\rangle_2 + b_3\langle G_x\rangle_3 \\ 2b_2\langle G_x\rangle_4 + b_3\langle G_x\rangle_5 \\ b_2\langle G_x\rangle_5 + 2b_3\langle G_x\rangle_6 \end{bmatrix} \delta^e \qquad (2.50(a))$$

$$= \frac{1}{2A} \gamma \; X \; \delta^e$$

$$\beta_{y,y} = \frac{1}{2A}(1,\xi,\eta) \begin{bmatrix} c_2\langle G_y\rangle_2 + c_3\langle G_y\rangle_3 \\ 2c_2\langle G_y\rangle_4 + c_3\langle G_y\rangle_5 \\ c_2\langle G_y\rangle_5 + 2c_3\langle G_y\rangle_6 \end{bmatrix} \delta^e \qquad (2.50(b))$$

$$= \frac{1}{2A}\gamma\, Y\, \delta^e$$

$$\beta_{x,y} + \beta_{y,x} =$$

$$\frac{1}{2A} \begin{bmatrix} c_2\langle G_x\rangle_2 + c_3\langle G_x\rangle_3 + b_2\langle G_y\rangle_2 + b_3\langle G_y\rangle_3 \\ 2c_2\langle G_x\rangle_4 + c_3\langle G_x\rangle_5 + 2b_2\langle G_y\rangle_4 + b_3\langle G_y\rangle_5 \\ c_2\langle G_x\rangle_5 + 2c_3\langle G_x\rangle_6 + b_2\langle G_y\rangle_5 + 2b_3\langle G_y\rangle_6 \end{bmatrix} \delta^e \qquad (2.50(c))$$

$$= \frac{1}{2A}\gamma\, Z\, \delta^e$$

where $\langle G_x\rangle_i$ and $\langle G_y\rangle_i$ represent the i^{th} rows of the matrices G_x and G_y respectively.

Substituting equations (2.50) into equations (2.10) and carrying out the area integral in equation (2.20) of the type :

$$\int_0^1 \int_0^{1-\xi} \xi^m \eta^n \, d\xi d\eta = \frac{m!n!}{(m+n+2)!}$$

the strain energy can be derived as :

$$U = \tfrac{1}{2}\, \delta^{eT} K\, \delta^e$$

K is the element stiffness matrix in global coordinates and is :

$$K = \frac{1}{2A} \begin{bmatrix} D_{11} R & D_{12} R & D_{13} R \\ D_{21} R & D_{22} R & D_{23} R \\ D_{31} R & D_{32} R & D_{33} R \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \qquad (2.51)$$

with :

$$R = \frac{1}{24} \begin{bmatrix} 12 & 4 & 4 \\ 4 & 2 & 1 \\ 4 & 1 & 2 \end{bmatrix} \qquad (2.52)$$

The terms of the material matrix D is defined by equation (2.16)..

The FORTRAN coding for this formulation is presented in reference [9]. Once the nodal displacements in global coordinates have been calculated, the bending moments at any point on the element follows from :

$$M = \frac{1}{2A} (1, \xi, \eta) \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \delta^e \qquad (2.53)$$

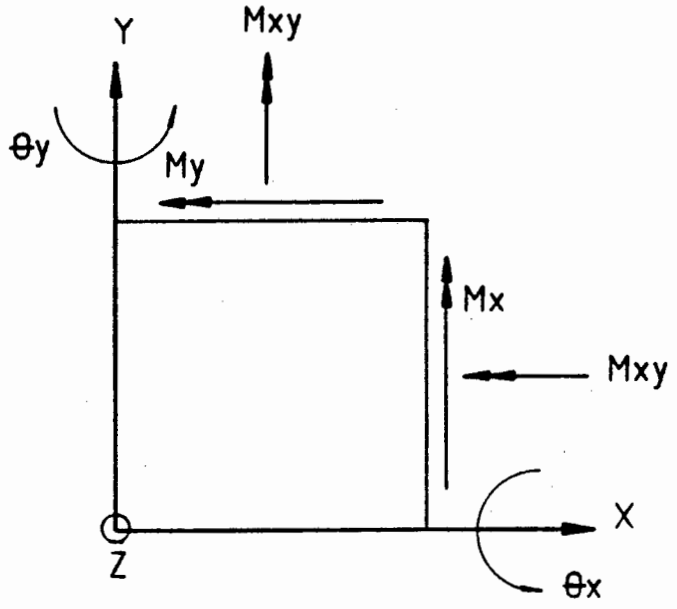


Figure 2.6 : Bending moments M_x , M_y and M_{xy}

2.4 Plane stress element

2.4.1 Introduction

The three noded triangular bending element dictates the same configuration for the plane stress element. The simple constant strain triangle (CST) element is still extensively used to describe this membrane action. However, the convergence of the CST element is slow compared to that of the DKT element and seems to detract from the overall performance of the element. Attempts to improve the membrane action of the element has, to date, not been particularly successful.

Olson and Bearden [10], formulate a plane stress element with two perpendicular displacements and an in-plane rotation per node. They conclude, however, that the stress distribution within the element is quite erratic and that the element is not capable of representing the constant stress states exactly (i.e. the formulation is incomplete).

The degenerate linear strain triangle with three degrees of freedom per node as presented by Carpenter et al [11] out performs the above formulation, but unfortunately only if a reduced integration scheme is used. Reduced integration introduces zero energy modes with the result that the element is not totally reliable under all conditions.

Talbot and Dhatt [12] concludes that for shell problems with bending dominated behaviour, superposition of the DKT and CST elements is probably the most efficient. We are concerned with the behaviour of prestressed flat slabs where the membrane action is solely due to the application of the in-plane prestressing component. The bending component most certainly dominates the behaviour of this type of plate. In-plane stresses are normally quite well distributed and can be modelled with sufficient accuracy by the CST element.

2.4.2 Formulation of the constant strain triangle

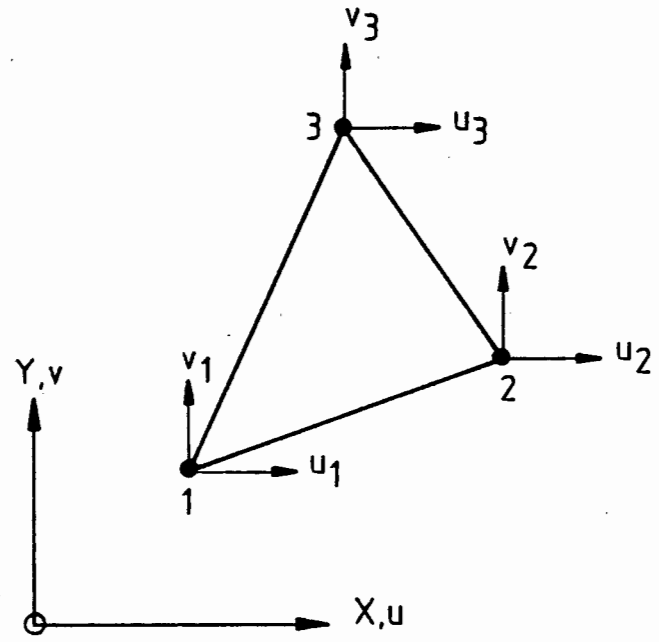


Figure 2.7 : Nodal displacements for the constant strain triangle

The assumptions for plane stress are :

$$u = u(x,y)$$
$$v = v(x,y) \qquad (2.54)$$
$$\sigma_z = \tau_{zx} = \tau_{zy} = 0$$

For a linear expansion of displacements (i.e. constant strain) we can write :

$$u(x,y) = [1-\xi-\eta \ , \xi \ , \eta] \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \qquad (2.55)$$
$$= \Phi \, U^e$$

$$v(x,y) = [1-\xi-\eta \ , \xi \ , \eta] \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \qquad (2.56)$$
$$= \Phi \, V^e$$

where ξ and η are area coordinates.

The strain displacements relationship for plane stress is :

$$\varepsilon_x = u_{,x}(x,y) = \Phi_{,x} \, U^e \qquad (2.57(a))$$

$$\varepsilon_y = v_{,y}(x,y) = \Phi_{,y} \, V^e \qquad (2.57(b))$$

$$\gamma_{xy} = u_{,y}(x,y) + v_{,x}(x,y)$$
$$= \Phi_{,y} \, U^e + \Phi_{,x} \, V^e \qquad (2.57(c))$$

or :

$$\varepsilon = \begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{bmatrix} = \begin{bmatrix} \Phi_{,x} & 0 \\ 0 & \Phi_{,y} \\ \Phi_{,x} & \Phi_{,y} \end{bmatrix} \begin{bmatrix} U^e \\ V^e \end{bmatrix} = B\,\delta^e \qquad (2.58)$$

Noting (2.49) we have :

$$\Phi_{,x} = \frac{1}{2A}(b_1 , b_2 , b_3)$$

$$\Phi_{,y} = \frac{1}{2A}(c_1 , c_2 , c_3) \qquad (2.59)$$

where $b_i = (y_j - y_k)$ and $c_i = (x_k - x_j)$, with (i,j,k) an even permutation of (1,2,3), i.e. :

$$\begin{array}{lll} i = 1 & j = 2 & k = 3 \\ i = 1 & j = 3 & k = 1 \\ i = 3 & j = 1 & k = 2 \end{array}$$

Stress strain relationships for a homogeneous plate are :

$$\sigma = \begin{bmatrix} \sigma_x \\ \sigma_y \\ \sigma_{xy} \end{bmatrix} = D\,\varepsilon \qquad (2.60)$$

For an isotropic material and constant plate thickness we have :

$$D = \int_{-\frac{h}{2}}^{\frac{h}{2}} D(z)\, dz$$

$$= \begin{bmatrix} D_{11} & D_{12} & D_{13} \\ D_{21} & D_{22} & D_{23} \\ D_{31} & D_{32} & D_{33} \end{bmatrix} \tag{2.61}$$

$$= \frac{E}{(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{(1-\nu)}{2} \end{bmatrix}$$

The strain energy for the plate is given by :

$$U = \tfrac{1}{2} \int_A \varepsilon^T D\, \varepsilon\, dA \tag{2.62}$$

Substituting (2.58) in (2.62) we have :

$$U = \tfrac{1}{2}\, \delta^{eT} K\, \delta^e \tag{2.63}$$

K is the stiffness matrix of the element and is :

$$K = \int_A B^T DB\, dA$$

$$= \begin{bmatrix} K_{uu} & K_{uv} \\ K_{uv}^T & K_{vv} \end{bmatrix} \tag{2.64}$$

where :

$$K_{uu} = \int_A [D_{11}\ \Phi^T_{,x}\ \Phi_{,x} + D_{13}\ (\Phi^T_{,x}\ \Phi_{,y} + \Phi^T_{,y}\ \Phi_{,x}) +$$

$$D_{33}\ \Phi^T_{,y}\ \Phi_{,y}]\ dA$$

$$K_{uv} = \int_A [D_{12}\ \Phi^T_{,x}\ \Phi_{,y} + D_{13}\ \Phi^T_{,x}\ \Phi_{,x} + D_{23}\ \Phi^T_{,y}\ \Phi_{,y}$$

$$+ D_{33}\ \Phi^T_{,y}\ \Phi_{,x}]\ dA \qquad (2.65)$$

$$K_{vv} = \int_A [D_{22}\ \Phi^T_{,y}\ \Phi_{,y} + D_{23}\ (\Phi^T_{,x}\ \Phi_{,y} + \Phi^T_{,y}\ \Phi_{,x})$$

$$+ D_{33}\ \Phi^T_{,x}\ \Phi_{,x}]\ dA$$

Carrying out the area integrals in equations (2.65) of the type :

$$\int_0^1 \int_0^{1-\xi} \xi^m\ \eta^n\ d\xi dq = \frac{m!n!}{(m+n+2)\ !}$$

we have the element stiffness matrix explicitly in global coordinates :

$$4AK_{uu}\Big|_{ij} = D_{11}\ b_i\ b_j + D_{33}\ c_i\ c_j + D_{13}\ (b_i\ c_j + c_i\ b_j)$$

$$4AK_{vv}\Big|_{ij} = D_{33}\ b_i\ b_j + D_{22}\ c_i\ c_j + D_{23}\ (b_i\ c_j + c_i\ b_j) \qquad (2.66)$$

$$4AK_{uv}\Big|_{ij} = D_{13}\ b_i\ b_j + D_{23}\ c_i\ c_j + D_{12}\ b_i\ c_j + D_{33}\ c_i\ b_j$$

where $b_i = (y_j - y_k)$ and $c_i = (x_k - x_j)$, with

(i,j,k) an even permutation of (1,2,3) i.e. :

i = 1	j = 2	k = 3
i = 2	j = 3	k = 1
i = 3	j = 1	k = 2

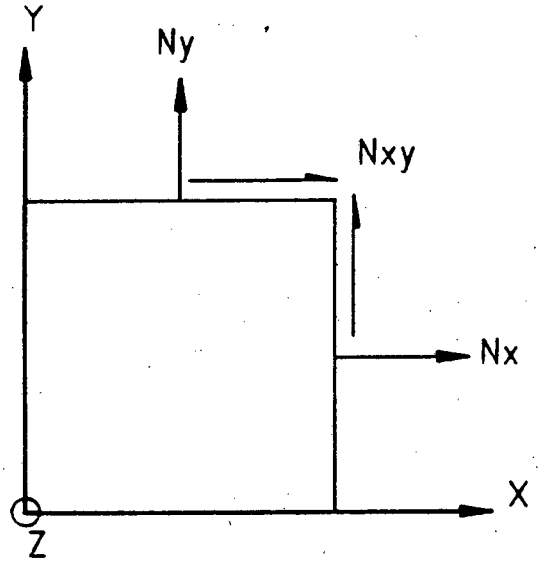


Figure 2.8 : In-plane components N_x , N_y and N_{xy}

As the strains are constant over the element, it follows that for a homogeneous isotropic plate the stresses are also constant. Stresses are calculated at the element centroid and follows from the nodal displacements in global coordinates :

$$\sigma = D_n \varepsilon$$

$$= \frac{1}{2A} D_n \begin{bmatrix} \sum_{i=1}^{3} b_i u_i \\ \sum_{i=1}^{3} c_i v_i \\ \sum_{i=1}^{3} (c_i u_i + b_i v_i) \end{bmatrix} \qquad (2.67)$$

where $D_n = \frac{1}{h} D$

2.5 Total element stiffness matrix

The total element stiffness matrix is formulated by superimposing the plane stress membrane stiffness K_m and the bending stiffness K_b i.e. :

$$K = \begin{bmatrix} K_m & \vdots & \\ \cdots & \vdots & \cdots \\ & \vdots & K_b \end{bmatrix} \qquad (2.68)$$

CHAPTER 3

BUILDING PROGRAMS FOR MICRO-COMPUTERS

3.1 Introduction

Of the introduction of the IBM/PC to the market in 1981, Norton [15] says :

> 'other people, who had always thought of personal computers as toys, began to recognise [personal] computers for what they really are - the premier tool for working minds'.

These 'other people' referred to, aptly describes the majority of practising engineers. Hardware is, of course, only one half of the tool, the other being software.

Most of the finite element software currently available emanates from either universities, big industrial corporations or large computer bureau. These instances have invested heavily in a vast stock of sophisticated programs for mainframe and mini-computers. As these machines became more powerful, so did the software. Unfortunately, the software has grown so sophisticated and specialized that much of it is no longer of practical use in day to day structural analysis and design.

Because micro-computers suffer from some of the same limitations as early mainframes, software developers saw their arrival as a retrogressive step. Micro-computers were viewed as the poor man's alternative to a larger machine and building finite element programs for them as re-inventing the wheel. The result is that most micro-computer based finite element software today is merely a translation of the mainframe/mini computer equivalent. This type of software totally ignores some of the most powerful features of micro-computers. One only has to note the fantastic features available on the multitude of data base, word processing, spreadsheet, or games packages to realise what is missing and what potential there is.

With a few notable exceptions, the result of this approach to the development of finite element based software for micro-computers has been :

- Slow running, large capacity programs.
- Limited pre- and post-processing, especially in the sphere of graphics and design orientated support.
- Resistance to the finite element method from practising structural engineers.

In order to develop suitable finite element software for micro-computers then, it is necessary to study the machine on which the program is intended to run and then to build the program using that machine, within the limitations of the machine.

3.2 The machine

The current development was carried out on an IBM/PC XT in the belief that, because of the rocketing sales of these and compatible machines, they will be around for some time and that there will be a substantial demand for software. The concepts underlying the current research is, however, general and apply to a vast range of machines.

Successful micro-computer programming implies a clear understanding of the machine and its working parts (both hard and software). Some of the more powerful features of micro-computers stem from working parts that are totally inaccessible from high level languages. A thorough study was made of the following aspects of the IBM/PC class of computer [15-18] :

- Operating system (DOS)
- Micro processors (8086/8088 and 8087)
- ROM BIOS
- Peripheral devices : display screen
  keyboard
  disk drives

There are currently some micro-computers that can operate as virtual machines and could, therefore, accommodate very large programs. Some compilers and interpreters also allow the option of loading and unloading sections of a program as required, thus freeing space in-core in a simulated virtual mode. The greatest drawback of building a finite element program as a unit lies in its inflexibility, i.e. :

- The whole program must be re-run for each analysis.
- The user has little or no control over the flow of the analysis and can not make value judgements about the analysis as it proceeds.

Modules composing the whole must interface with each other. This interface occurs on some storage medium such as floppy- or fixed disks and is often comparatively time consuming. The advantage in this is that the analysis can be terminated at any convenient point and then resumed from there at some other time. Careful programming is required to optimise this phase.

The program structure and functioning of the individual modules that make up "PRESLAB" are discussed in the following chapters.

3.4 Numerical methods for sets of linear equations

3.4.1 The equations

The current development is based on the linear elastic analysis of flat plates. For this type of analysis, the structure stiffness matrix is known to be sparsely populated, square, symmetric, positive definite and usually banded.

The sets of linear algebraic equations resulting from the finite element analysis of structures can be represented in matrix form as :

$$K\delta = F \qquad (3.1)$$

where K is the structure stiffness (coefficient) matrix, δ is the deflection vector and F is the loadvector.

Direct solutions based on Gauss elimination are the most efficient for solving the sets of linear equations in (3.1). The most suitable algorithms are those which decompose the coefficient matrix into a lower/upper triangular form [1,2,5,28-30]. This category of algorithm includes Crout reduction and the Cholesky method.

3.4.2 Crout reduction and the Cholesky method

Detailed discussion of these two algorithms abound in the literature, [1,2,5,28-30] and will not be repeated here. Symbolically, the solution of the system of linear equations in (3.1) proceeds as follows :

a) Crout reduction :

- Decomposition : $K = U^T D U$ (3.2(a))
- Forward reduction : $U^T y = F$ (3.2(b))
- Back substitution : $DU\delta = y$ (3.2(c))

b) The Cholesky method :

- Decomposition : $K = U^T U$ (3.3(a))
- Forward reduction : $U^T y = F$ (3.3(b))
- Back substitution : $U\delta = y$ (3.3(c))

Crout reduction has the advantage of being able to accommodate non-positive definite coefficient matrices. The structure stiffness matrix only becomes non-positive definite if the structure can undergo rigid body movements or if the material matrix is non-positive definite. In a linear elastic analysis, this means that the structure is not properly restrained or that the material properties are defined incorrectly (i.e. Poisson's ratio > 0.5). For this type of analysis then, Crout reduction has no material advantage over the Cholesky method.

The Cholesky method, on the other hand, requires less operations, data storage space and data transferals from auxiliary storage in an out-of-core solution algorithm. This method was used in "PRESLAB".

3.4.3 Solving the deflection vector

The Cholesky method is ideally suited to modular programming. Solution of the equations in (3.1) is divided into the following two distinctly separate programs (termed modules) :

a) Module 1 :

Decomposition of the coefficient matrix to the form :

$$K = U^T U \tag{3.4}$$

The decomposed form of the coefficient matrix (U) is stored on auxiliary storage. This step of the solution is only carried out once for a particular structure.

b) Module 2 :

Forward reduction and back substitution of the load-vectors to obtain the nodal displacements. This module uses the decomposed form of the coefficient matrix above. Loadcases can be considered at any stage and in any order without having to set up and decompose the structure stiffness matrix again.

A detailed discussion of the solver follows in Chapter 6.

CHAPTER 4

THE PROGRAM

4.1 Macro flow diagram of the program

The macro flow diagram in Figure 4.1 illustrates the various modules that make up "PRESLAB". Each module is an independent program that interfaces with other modules through files of data on auxiliary storage.

The program can be divided into four sections, i.e. :

- Super-element section.
- System section.
- Load case section.
- Post-processing section.

Logical flow of the program modules is regulated by a series of control flags. No module can be run out of sequence and any material changes to the primary data will re-set the appropriate flags.

4.2 The editors

Inquiry/response type of editors are normally encountered on micro-computer based finite element packages and are termed 'interactive pre-processors'. In terms of micro-computer based software, these editors are totally unacceptable and only indicates ignorance of this type of machine.

The three modules, SUPRED, EDSYS and LOADED are on-screen data editors and form the backbone of the user/machine interface. They serve as a powerful means of quickly producing and editing the vast amount of data required for a finite element analysis. In reality, the editors are sophisticated links between the keyboard, video display and auxiliary storage that also anticipate user responses to some extent.

The basic philosophy behind the development of the editors was such that they must :

- be easy to understand
- be easy to use
- be impossible to foul or snag
- be menu driven
- operate on tables of data
- use default values where possible
- generate sequences of data (FROM-TO-STEP mode)
- generate data automatically whenever possible
- allow data changes anywhere in the tables
- allow data additions and deletions anywhere in the tables
- do error trapping
- have audio visual prompts and responses
- be adaptable.

Because of the complex nature and size of these modules, it was not possible to build data logic checking routines into the editors. These checks are carried out and reported in the module following the editor. No basic data is removed from disk and can be edited at any stage. Checks for data inconsistencies are, however, carried out as a matter of course.

4.3 Super-element section

This section allows for the rapid generation of elements through the use of super-elements. The algorithm is discussed in Chapter 5. The super-element section consists of the following modules :

- SUPRED All the data required to define the super-elements is entered through this editor. Super-elements are defined by four corner nodes in either rectangular or polar co-ordinates. The number of elements to be generated for each super-element is also prescribed.

- PLOTSUP This plotting routine allows the user to view the super-elements graphically before generating the elements. Super-element data can, of course, be edited at any time. Plotting routines are more fully described in Section 4.7.

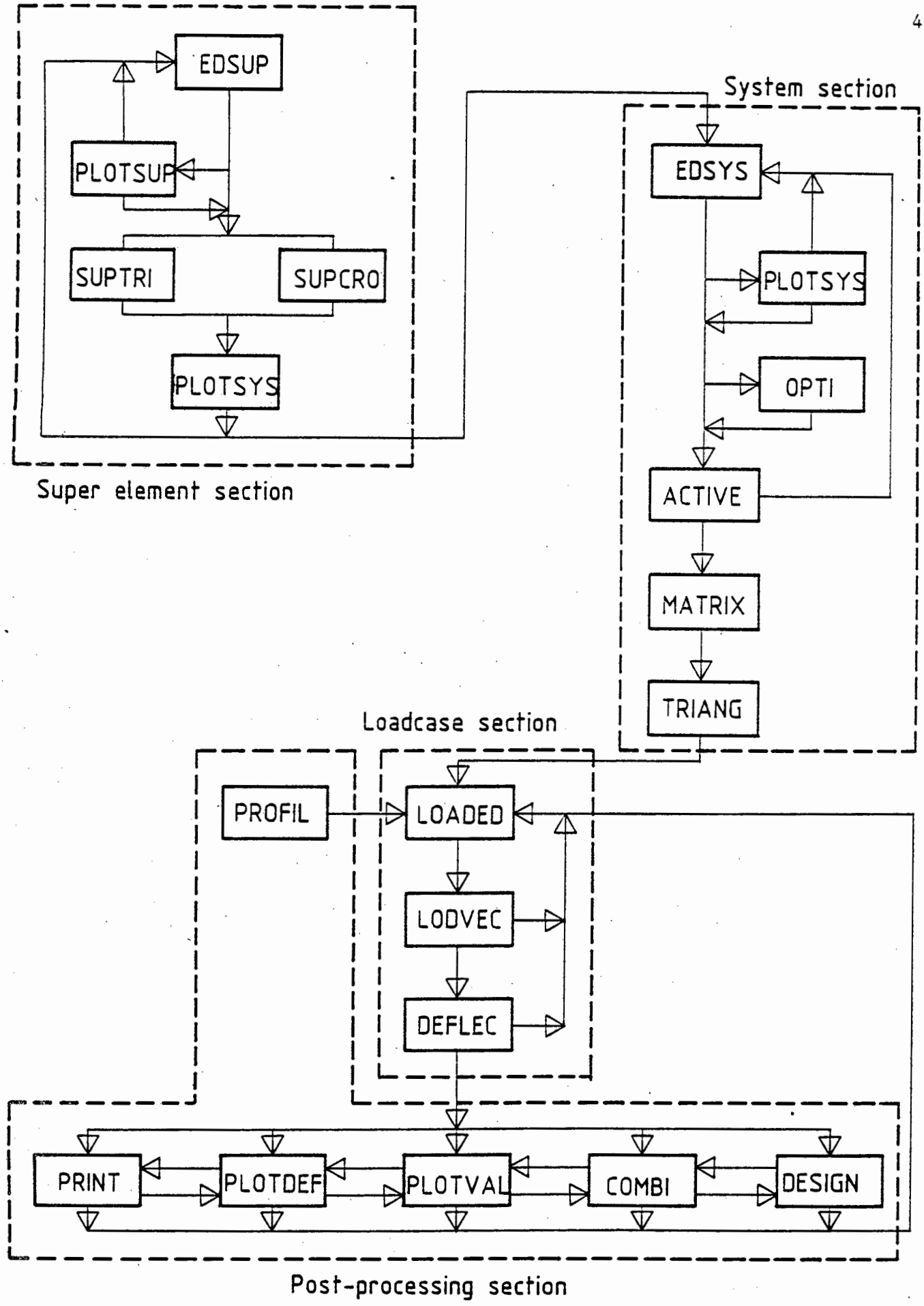


Figure 4.1 : Macro flow diagram of the program PRESLAB

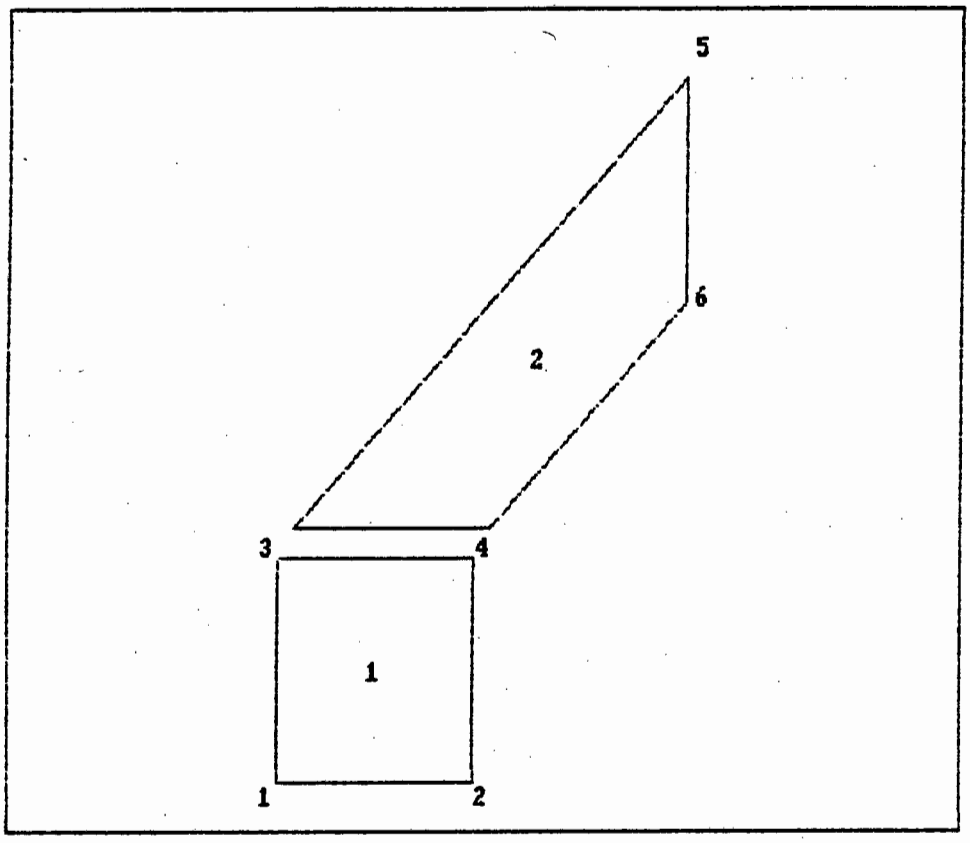


Figure 4.2 : Typical super-element model

- SUPTRI The actual elements, their incidences and the nodal coordinates are generated here. Elements are first generated as four noded quadrilaterals and then triangulated across the shortest diagonal. Also included in this module is a bandwidth reduction scheme (cf. Chapter 5) and an option to print the generated data.

- SUPCRO This module is exactly the same as SUPTRI except that the quadrilateral elements are traingulated across both diagonals. Four triangular elements are generated per quadrilateral resulting in a more symmetrical pattern of elements.

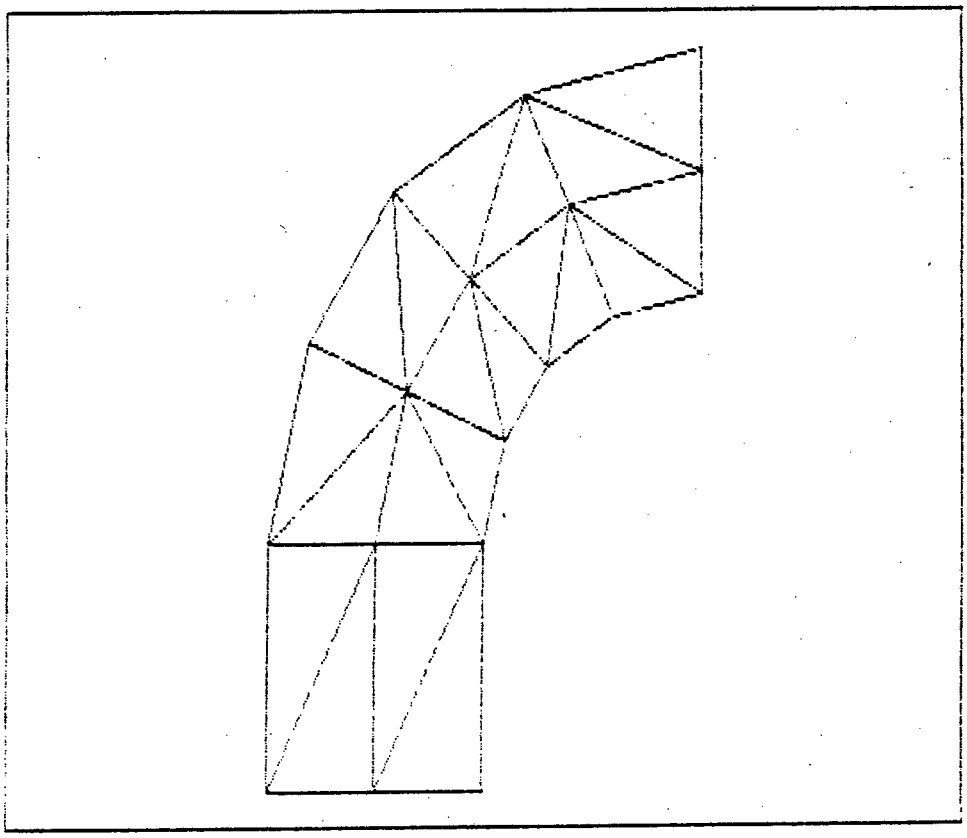


Figure 4.3 : Typical model generated by SUPTRI

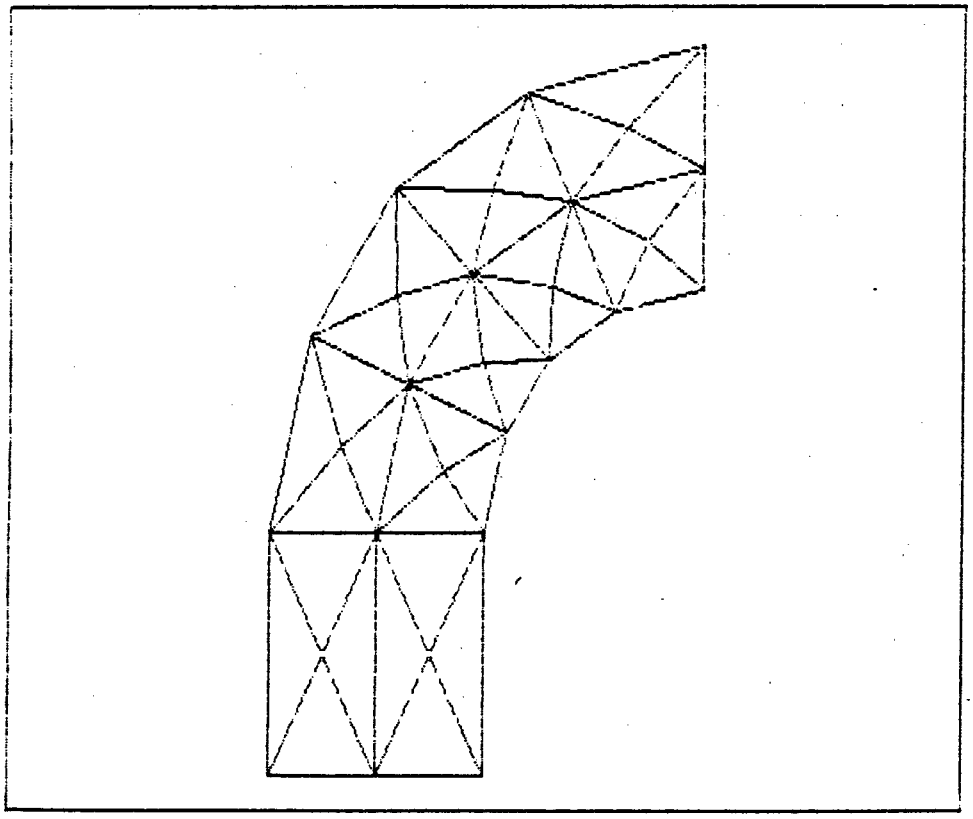


Figure 4.4 : Typical model generated by SUPCRO

The super-element section is really a utility to the system section and is not essential to the overall functioning of the programs. Material data or nodal fixities are not generated in this section.

PLOTSYS displays the generated finite element mesh graphically and is included as a link between the super-element and system sections.

4.4 System section

Most of the numerical computation of the solution is carried out in this section which consists of the following modules :

- EDSYS — All the data required for a finite element analysis excluding the load data is entered through this editor. Data generated by the super-element section is automatically retrieved. Material data and nodal fixities are not generated by the super-element section and need to be specified through this editor.

- PLOTSYS — The system is displayed on the screen using this plotting module. Any discrepancies in the finite element mesh can easily be spotted and rectified before the analysis continues. Plotting routines are more fully described in Section 4.7.

- OPTI — This module is optional and performs bandwidth reduction using the Collins [31] algorithm. Nodal renumbering resulting from this scheme is transparent to the user. The algorithm is discussed in Chapter 5.

- ACTIVE — A number of essential utilities are performed by this module i.e. :

  a) Input data is checked for errors. All errors are reported and the program halts execution if any are detected. Possible error include :

- Zero or less nodes and/or elements
- Number of nodes exceed 3 x number of elements
- Number of fixed nodes out of range
- Number of materials out of range
- Nodes with identical coordinates
- Unspecified material numbers
- Node numbers out of range
- Duplicate node numbers
- Nodes not associated with elements
- Fixed nodes out of range
- Fixed nodes with no deleted degrees of freedom
- Fixed nodes defined more than once
- Zero elemental area
- Elements numbered clockwise
- Bandwidth too large for problem to be solved.

b) The "skyline" of the structure stiffness matrix is determined and stored as an array of the addresses of the leading diagonal terms in the matrix (the DA array). This efficient method of storing the structure stiffness matrix as a one dimensional array is described in Chapter 6.

c) The program was built to operate in- or out-of-core depending on the size of the problem. For large problems the assembly of the structure stiffness matrix must also occur out-of-core. In order to minimise I/O during the assembly of the structure stiffness matrix, the positions of each element stiffness matrix in the structure stiffness matrix is determined. Elements are then arranged in ascending order of appearance using the efficient "bubble" sorting routine and assembly of the structure stiffness matrix is divided into consecutive blocks that can fit into core.

d) A vector is set up that identifies each deleted degree of freedom. This vector is utilised in the solution of the linear equations and in recovering the reactions.

e) Space is reserved on auxiliary storage for the stiffness matrix and reaction equations.

f) Coordinates of each element midpoint is calculated and the outline of the model is determined. This data is stored on auxiliary storage to be used in plotting and printing routines.

g) An option that prints all the input data.

- MATRIX The structure stiffness matrix is assembled for all degrees of freedom. This matrix is assembled using the DA array and assembly order determined in the ACTIVE module. Reaction equations are also recovered and stored on auxiliary storage.

- TRIANG This module triangulates the structure stiffness matrix by the Cholesky method. An in depth discussion of the solver follows in Chapter 6. Suffice it to say that the decomposed stiffness matrix is stored on auxiliary storage. Triangulation occurs in-core or out-of-core depending on the size of the problem.

4.5 Load case section

The load case section is the work horse of the design engineer. Load cases are defined, load vectors set up and deflection- and stress resultants calculated.

The system section represents most of the computational effort in the finite element method. Compared to this, the load case section is a lot faster. Three modules make up the load case section, and are :

- LOADED Load cases are defined and the loads specified using this editor. Self weight, area-, line- and nodal loads can be specified. Self weight, area- and line loads can act in any direction as defined by three components of a user supplied unit vector. Self weight is calculated from the element volume and weight density. Nodal loads include

all five degrees of freedom (three point loads and two moments). Any number of load cases can be specified in any order.

- LODVEC Input data of all new load cases is checked for errors. Possible errors include :

  a) Area loads :

     i) Element numbers out of range.

  b) Line loads :

     i) Node numbers out of range.
     ii) Pair of line load node numbers the same.
     iii) Pair of line load node numbers not on the same element (i.e. not in line).

  c) Nodal loads :

     i) Node numbers out of range.

  If any errors are detected, they are reported and the load vector for that load case is not assembled. All other load vectors are assembled using the static resultants of the applied loads as nodal loads and stored on auxiliary storage. The load data for all assembled load vectors is removed from disk to save space.

  A print option allows the load data and diagnostics to be printed.

- DEFLEC Displacement vectors are calculated using the decomposed upper triangular form of the stiffness matrix and the load vector on auxiliary storage. Displacement vectors are stored in place of the load vectors. A detailed discussion of the algorithm follows in Chapter 6.

  Stress resultants are calculated at each element centroid and stored on auxiliary storage.

Load cases can be added or amended at any stage.

4.6 Post processing section

The post processing section contains the creative tools of the design engineer and allows him to :

- print, view and combine load cases with relative ease,
- display deflected shapes and vector plots and
- design untensioned reinforcement.

Post processing does not offer a self contained ("black box") solution to the design of prestressed flat slabs, but gives the designer insight into the problem and guides his decisions.

The modules that make up this section are :

- PROFIL This module is really a pre-processor for the load case section. Loading due to prestressing cables and their profiles are calculated and presented graphically. Discussion of the theoretical background follows in Chapter 7. Various cable profiles can be tried and assessed relatively quickly.

- PRINT Deflections, reactions, moments and in-plane forces for an existing load case can be printed. The designer has the choice of which results to print, thus saving time and unnecessary print out.

- PLOTS Two plotting routines display the results of an analysis graphically. PLOTDEF displays the deflected shape of the structure and PLOTVAL displays vector plots of the moments and in-plane forces. Plotting routines are discussed in Section 4.7.

- COMBI Any number of existing load cases can be combined to form new load cases. Load factors are specified and applied to the basic load cases in the combination. This module

allows a quick method of exploring a number of options especially when applying prestressing. The design example in Chapter 10 illustrates this clearly.

- DESIG This module aids the design of untensioned steel for prestressed flat slabs. Design theory is discussed in Chapter 8. For any load case, one of the following three options can be printed for each element :

  a) Principal stresses and directions in the top and bottom fibres of the slab.

  b) Wood/Armer [40,41] design moments and in-plane forces in any two directions.

  c) Recommended [48] untensioned steel areas per unit length for any two reinforcing directions.

In order to use these modules, the design engineer must have a clear understanding of the principles of prestressed concrete and the design of flat slabs under a combination of bending and in-plane forces. The modules will aid him in visualising the behaviour of a specific problem under various conditions of loading and prestress and guide his decisions regarding the design of the slab. The design program can also make recommendations about the amount of untensioned steel required for specific load cases.

4.7 Plotting routines

'Programs are often judged by their display quality and visual design alone' [16].

Four plotting modules form the basis of the visual display interface with the user. Visual display is fundamental to the pre- and post processing philosophy followed throughout the development.

On the pre-processing side, PLOTSUP and PLOTSYS display the super-element and system models of the problem respectively. The model is constructed by plotting each element. These plots aid in describing the finite element model quickly and accurately.

- ZOOM-IN This toggle allows the user to change the scale of the plot in order to view the model in more detail. Detail areas are selected by means of a "rubber banded" rectangle.

- PANN Once the scale of a plot has been changed, it is often useful to view other areas of the model in the same scale. The viewing window can be "panned" across the model in any direction.

- NORMAL This toggle re-sets the scale of the plot and displays the whole model.

- DONE Execution of the program terminates.

PLOTDEF and PLOTVAL are part of the post-processing section of the program and display the deflected shape and vector plots of the results respectively.

PLOTDEF is useful when using the "load balancing" method in the design of prestressed flat slabs. This concept is illustrated in Chapter 10. The load case for which the deflected shape must be displayed is selected from all the available load cases.

Toggles on the "soft" keys allow the following modifications to the plot :

- SCALE The magnification factor for deflections is defined through this toggle.

- PERSPECT This toggle allows the model to be displayed in any of the four normal isometric projections or in plan. Limiting the choice of projection simplifies the programming requirements. Plots can, therefore, be generated very rapidly while still displaying sufficient information clearly.

- SYSTEM This toggle turns the plot of the un-deflected shape of the model on and off. When on, the un-deflected shape is superimposed on the deflected shape in a different line style (i.e. dotted lines).

- ZOOM-IN This toggle allows the user to view selective portions of the plot only and works in the same way as described previously.

- PANN The viewing window can be "panned" across the model in any direction.

- NEXT The program returns to the load case selection menu.

- DONE Execution of the program terminates.

PLOTVAL displays vector plots of the results of a finite element analysis. Only the outline of the structure is displayed and vectors are centred about element midpoints. Negative values are plotted as double lines. Three menus control the type of results plotted, i.e. :

- MENU 1 Select load case to plot.

- MENU 2 Select either moments or in-plane forces.

- MENU 3 Select X and Y values, XY values or principal values.

Toggles on the "soft" keys allow the following modifications to the plot :

- SCALE The magnification factor for the vector plot is defined through this toggle.

- NORMAL This toggle resets the scale of the plot and displays the whole model.

- ZOOM-IN This toggle allows the user to view selective portions of the plot only and works in the same was as described previously.

- PANN The viewing window can be "panned" across the model in any direction.

- NEXT DIR The program returns to meny 3.

- NEXT TYP The program returns to menu 2.

- NEXT L/C The program returns to menu 1.

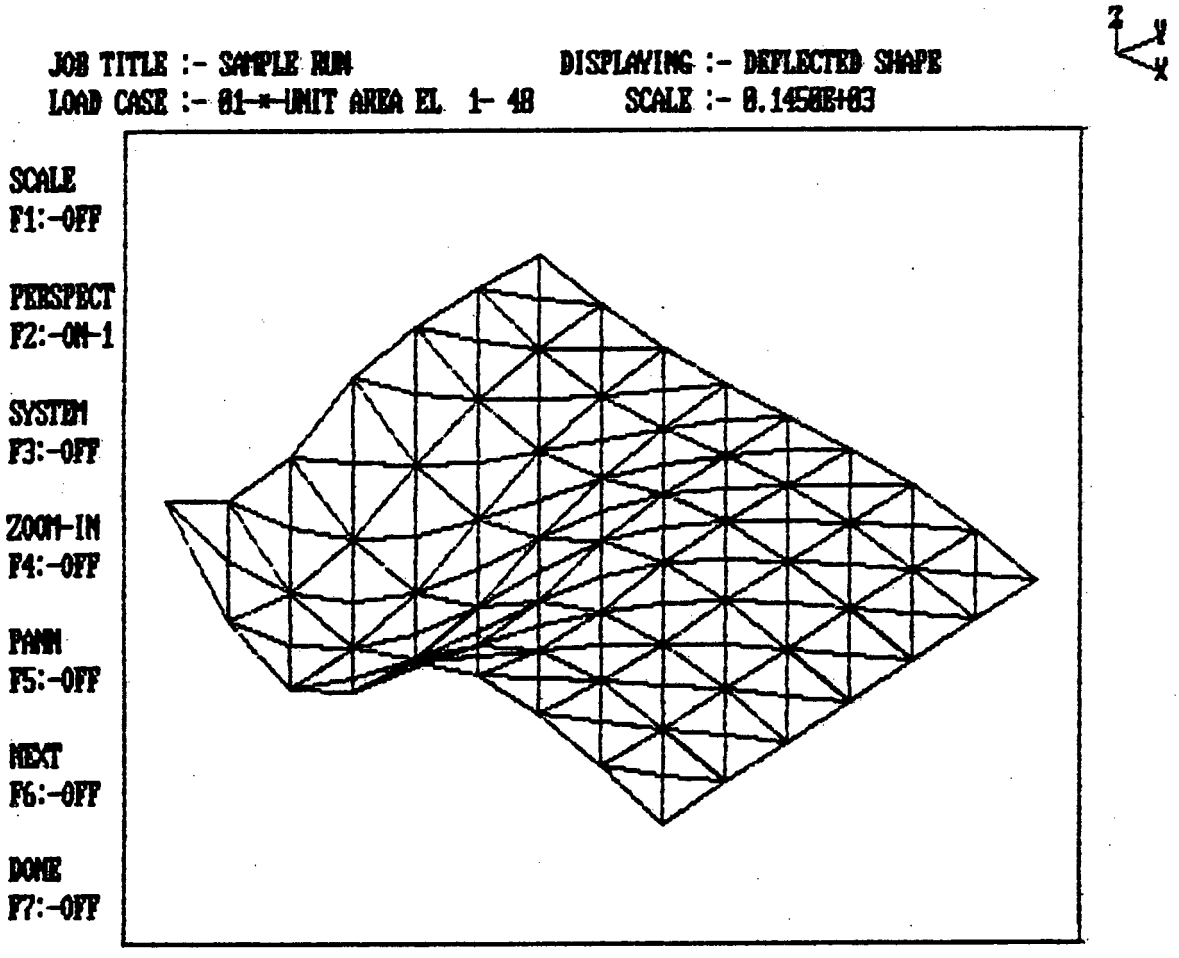


Figure 4.6 : Typical deflected shape plot

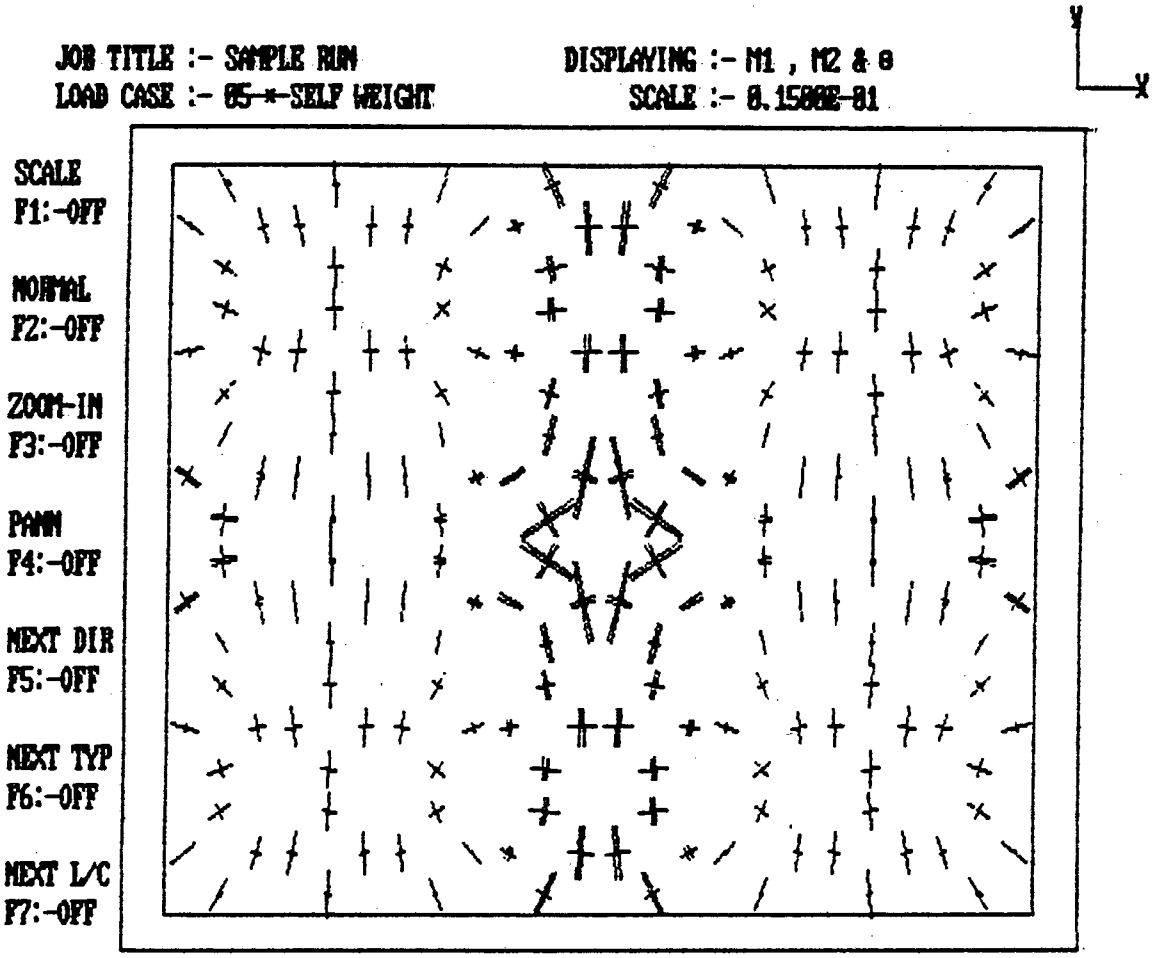


Figure 4.7 : Typical vector plot

4.8 Data files

The modules that make up the program interface with each other through data files on auxiliary storage. In this development the hard disk drive (or "winchester" drive) was used. Flexible disks can also be used, but are a lot slower and have far less capacity.

The operating system (DOS) allows two ways of storing data records on disks; sequential file records and random file records. Of these two, random file records have the fastest access time and require the least amount of space on disk. The disadvantage of random file records is that all records in a file must have the same size (in bytes). Random file records were used throughout this development.

Most disks are arranged in sectors, cylinders (or tracks in the case of flexible disks) and sides. The DOS operating system was designed to read and write standard 512 byte sectors. With this in mind, the modules were designed to do all I/O in records of 512 bytes or multiples thereof.

4.9 Numerical accuracy

Micro-computers offer a variety of formats in which numbers can be expressed and manipulated. These formats depend on the construction of the micro-processors and compiler support. In general though, the more accurate the numbers, the longer it takes to manipulate them and the more space they require.

In the current development, the highest degree of accuracy was used for calculations that produce intermediate results which could possibly affect the accuracy of the final solution. For all other values, the lowest feasible accuracy was used in order to optimize the space requirements of the program.

The use of the 8087 maths co-processor is deemed to be essential for developments of this nature. This co-processor can perform floating-point calculations in the neighbourhood of 10 to 50 times faster than the 8088 [18], with phenomenal numerical accuracy (80 bits). In trials done with the solution phase modules of the current development, run times were improved by a factor of at least four when using the 8087. The compilers used for the development, all support the 8087 co-processor.

4.10 Batch files

Batch files are a special form of command processing files. They contain a series of commands which the operating system will carry out as if they were entered at the keyboard. Two batch files can be constructed to aid the flow of the program.

The first, which is named SOLVE, runs the following programs in succession :

- ACTIVE
- MATRIX
- TRIANG

and the second, named EDLOD, runs the following programs :

- LOADED
- LODVEC
- DEFLEC

More complex batch files can be constructed to suit the needs of individual users.

CHAPTER 5

MESH GENERATION AND BANDWIDTH REDUCTION

5.1 Mesh generation

One of the first steps in building a finite element model of a physical problem is the sub-division of the domain into a finite number of elements. This step is often time consuming and tedious. Automatic generation of the elements reduces effort and the likelihood of errors. The accuracy of the solution may also increase because a computer generated mesh is more regular than one supplied by the user.

The mesh generation scheme used in the program was described by Zienkiewicz and Philips [32] and adapted by Durocher and Gasper [34]. Prenter [33] describes the mathematical basis of the problem in a clear and concise way.

Consider the parabolic super-element in Figure 5.1.

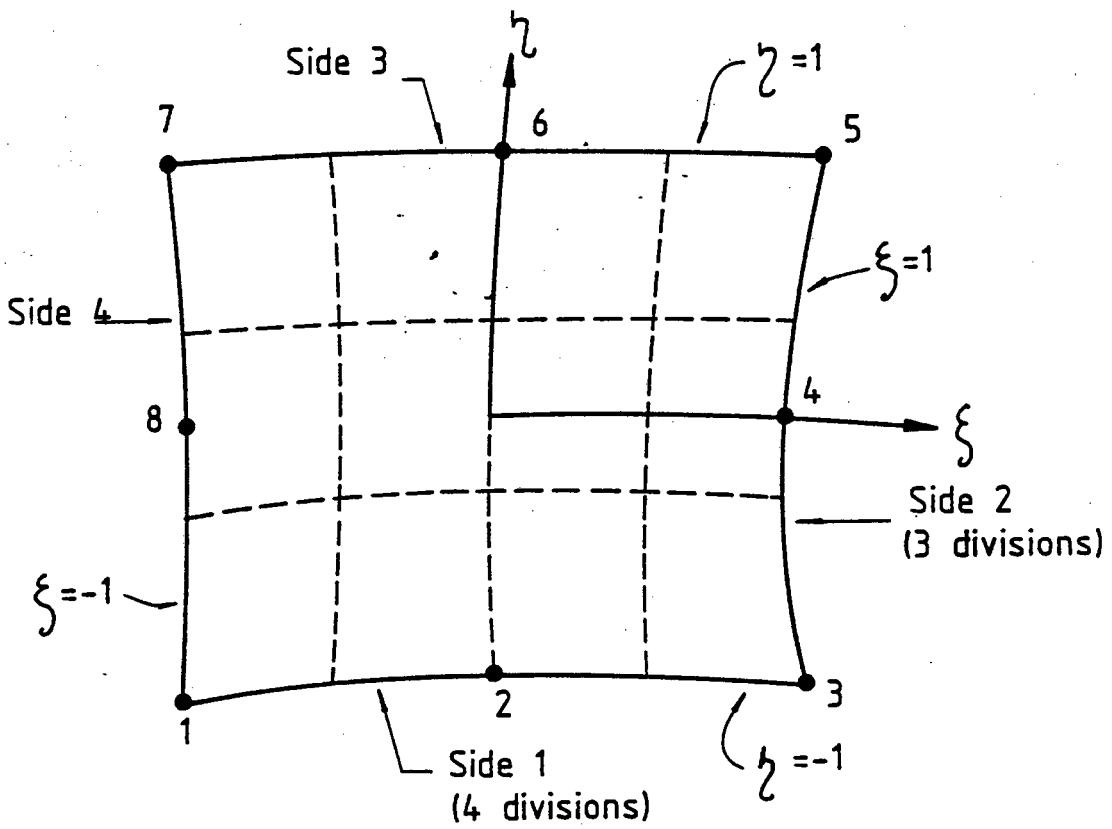


Figure 5.1 : Typical parabolic super-element

The x- and y-coordinates of any point on the super-element is related to the coordinates of the four corner and four midside nodes by :

$$x = \sum_{i=1}^{8} N_i (\xi,\eta)\, x_i \qquad (5.1(a))$$

$$y = \sum_{i=1}^{8} N_i (\xi,\eta)\, y_i \qquad (5.1(b))$$

where $N_i (\xi,\eta)$ are the shape functions and ξ and η are curvilinear coordinates. For a quadratic variation along the boundary, the shape functions are given by :

$$N_1 = 0.25\ (1-\xi)\ (1-\eta)\ (-\xi-\eta-1)$$

$$N_2 = 0.5\ \ (1-\xi)\ (1-\eta)$$

$$N_3 = 0.25\ (1+\xi)\ (1-\eta)\ (\xi-\eta-1)$$

$$N_4 = 0.5\ \ (1+\xi)\ (1-\eta)$$

$$N_5 = 0.25\ (1+\xi)\ (1+\eta)\ (\xi+\eta-1)$$

$$N_6 = 0.5\ \ (1-\xi)\ (1+\eta)$$

$$N_7 = 0.25\ (1-\xi)\ (1+\eta)\ (\xi-\eta-1)$$

$$N_8 = 0.5\ \ (1-\xi)\ (1-\eta)$$

Super-elements are described by four corner nodes in either rectangular or polar coordinates. A polar coordinate system is established by defining the cartesian coordinates of the midpoint. The coordinates of a node are then specified as a radius and an angle relative to the origin and local x-axis of the polar system respectively (Figure 5.2). Polar coordinates allow a means of modelling circular holes and boundaries.

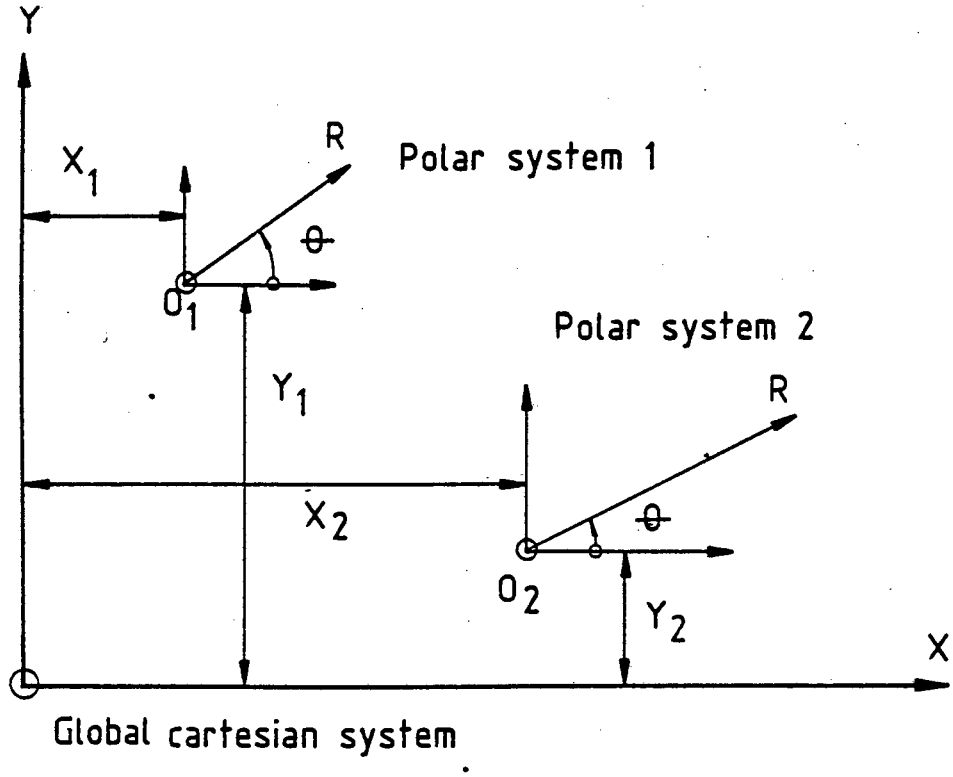


Figure 5.2 : Polar coordinate system

Midside nodes are established through interpolation. If the two end nodes of any side are described by the same polar system and are on the same radius, the midside node is interpolated using that same polar system. All other midside nodes are established by linear interpolation in cartesian coordinates.

Using equations (5.1) and (5.2) it is a simple task to generate a mesh with a specified number of sub-divisions in the ξ and η directions. Four corner nodes of a quadrilateral element are generated. These quadrilaterals are then triangulated across the shortest diagonal or across both diagonals (Figure 5.3).

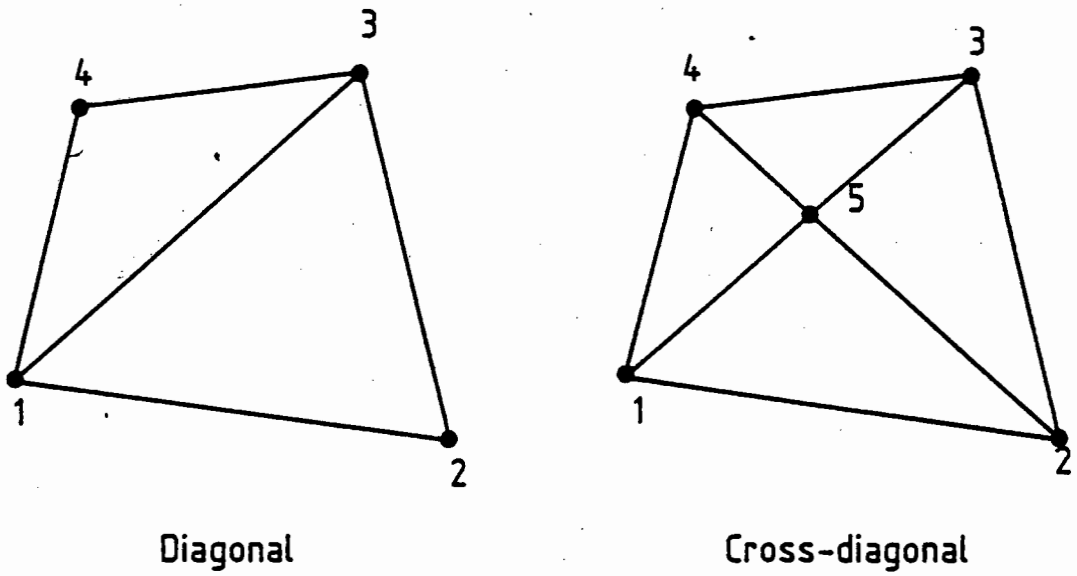


Figure 5.3 : Triangulation of quadrilateral elements

The super-element editor (SUPRED) maintains nodal compatibility of the generated mesh. Sub-division of a side of any super-element will automatically be reflected on all super-elements that are linked by common boundaries.

5.2 Bandwidth reduction

A 'banded' solution technique is used in "PRESLAB" to solve the sets of simultaneous linear equations resulting from the finite element analysis. To take full advantage of the solver, the structure stiffness matrix must be as narrowly banded as possible. The order in which the nodes are numbered affects the bandwidth of the structure stiffness matrix.

Several reasons can be raised as to why it is impractical to attempt an optimum numbering scheme manually, i.e. :

- Few practicing engineers have the experience, knowledge and insight to do optimum numbering manually.

- Changes in the design may necessitate the addition or deletion of nodes at inconvenient locations. This might mean a total renumbering of the nodes in the model to achieve a narrow bandwidth.

- Nodes generated through the super-element section are numbered arbitrarily and often results in a very large bandwidth.

The Collins [31] bandwidth reduction scheme is employed to improve the numbering of the nodal points. In the super-element section, nodal renumbering is automatic. The module OPTI allows the user to renumber the nodes of a model at any time. Nodal renumbering is transparent to the user.

The bandwidth of the structure stiffness matrix is proportional to the maximum difference in node numbers of the nodes in any one element in the model. Nodal renumbering must then attempt to minimise the difference in the node numbers of all the elements.

A list of all nodes related to any one node in the structure is set up. Any one node is related to any other node if they both appear in the same element. The list of associated nodes for the model in Figure 5.4 appears in Table 5.2.

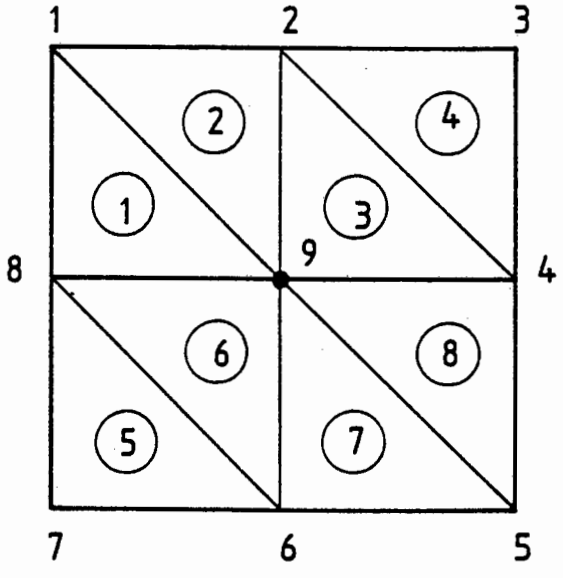


Figure 5.4 : Illustrative model

TABLE 5.2

NODE NO.	RELATED NODES
1	9,8,2
2	1,9,4,3
3	2,4
4	2,9,3,5
5	9,6,4
6	8,7,9,5
7	8,6
8	1,9,6,7
9	1,8,2,4,6,5

Starting with node 1, all nodes associated with it are renumbered consecutively. The maximum nodal difference, on the element level, measures the feasibility of the renumbering. If the maximum nodal difference exceeds that of the previous minimum, the nodal renumbering is abandoned and renumbering starts with the next node in the sequence. For a model with N nodes, N numbering schemes will be attempted (Figure 5.5).

The Collins algorithm does not renumber the nodes for a minimum bandwidth; it only attempts to reduce the bandwidth. Reliable results are achieved quickly.

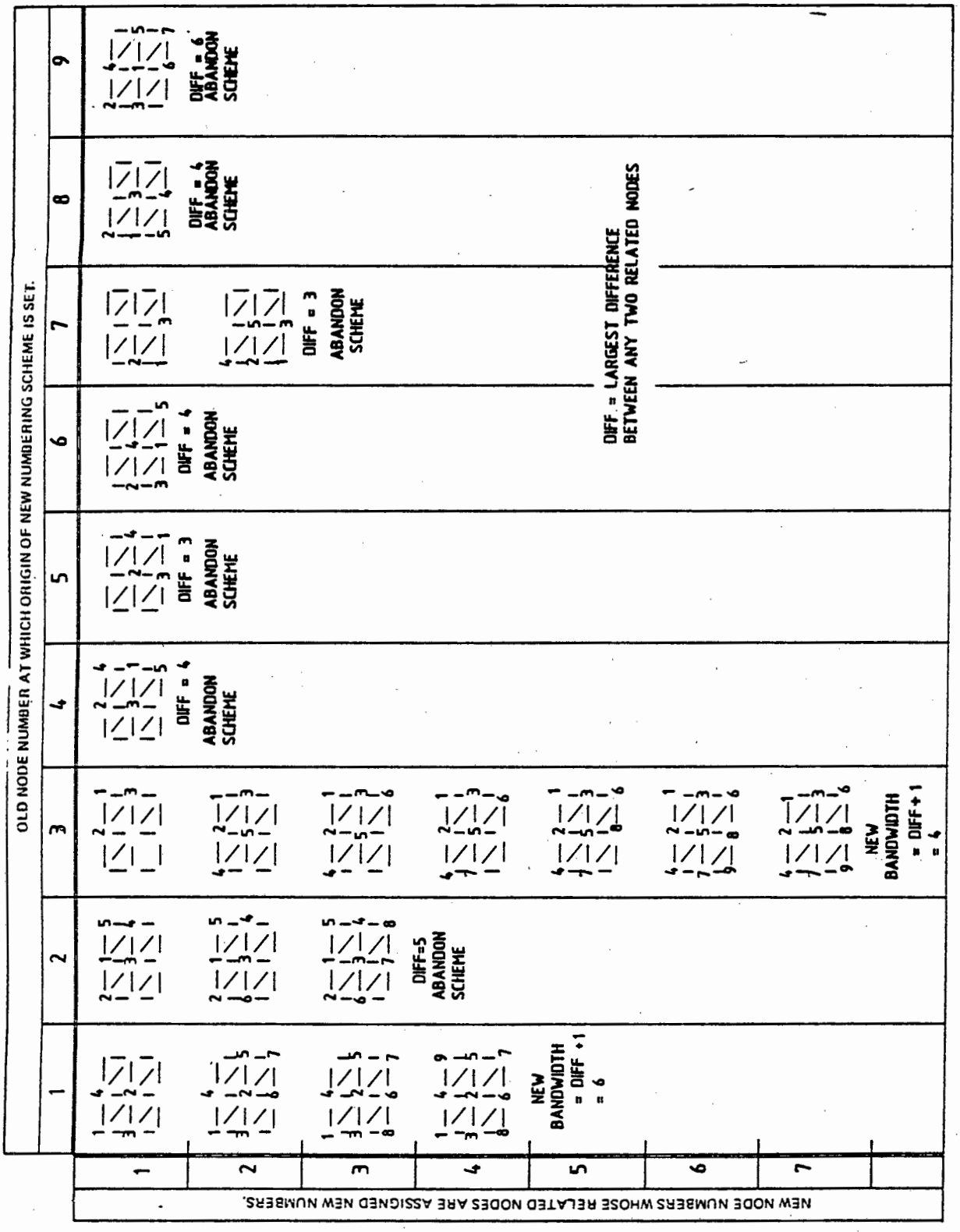


Figure 5.5 : Renumbering of nodes by the Collins algorithm

CHAPTER 6

SOLVER

6.1 Introduction

Micro-computers usually have some limitations on the amount of in-core storage available. A solver with an out-of-core capability is, therefore, essential. Previous research in this field has, on the whole, concentrated on minimising in-core requirements. However, it is clear that the optimal use of in-core memory is more in keeping with modern trends in hardware development.

The coefficient matrix, resulting from the linear elastic finite element analysis of structures, is sparse, symmetric and positive definite (cf. Chapter 5). These characteristics are effectively exploited in the development of efficient algorithms for the solution of large sets of simultaneous equations. The most well known and extensively used of these algorithms is the rigid block partitioned form of solution [24].

The current development improves on the rigid block partitioning algorithm. The improvements deal in general with the manner in which the coefficient matrix is partitioned.

6.2 Storage of the coefficient matrix and load vectors

The coefficient matrix is stored on auxiliary storage in the well known as "skyline" form [1]. Only terms above and including the diagonal and below the skyline are stored as a one-dimensional array. These terms are stored for each consecutive column (Figure 6.1).

The skyline storage scheme is more complex to set up and manipulate than the often used banded form, but is far more efficient in terms of data transfer and in-core memory requirements.

In order to identify specific elements in the coefficient matrix from the one-dimensional format of the skyline scheme, a diagonal address (DA) array is required. The DA array lists the addresses of the diagonal terms in the coefficient matrix (Figure 6.1.). Load vectors are stored consecutively for each load case.

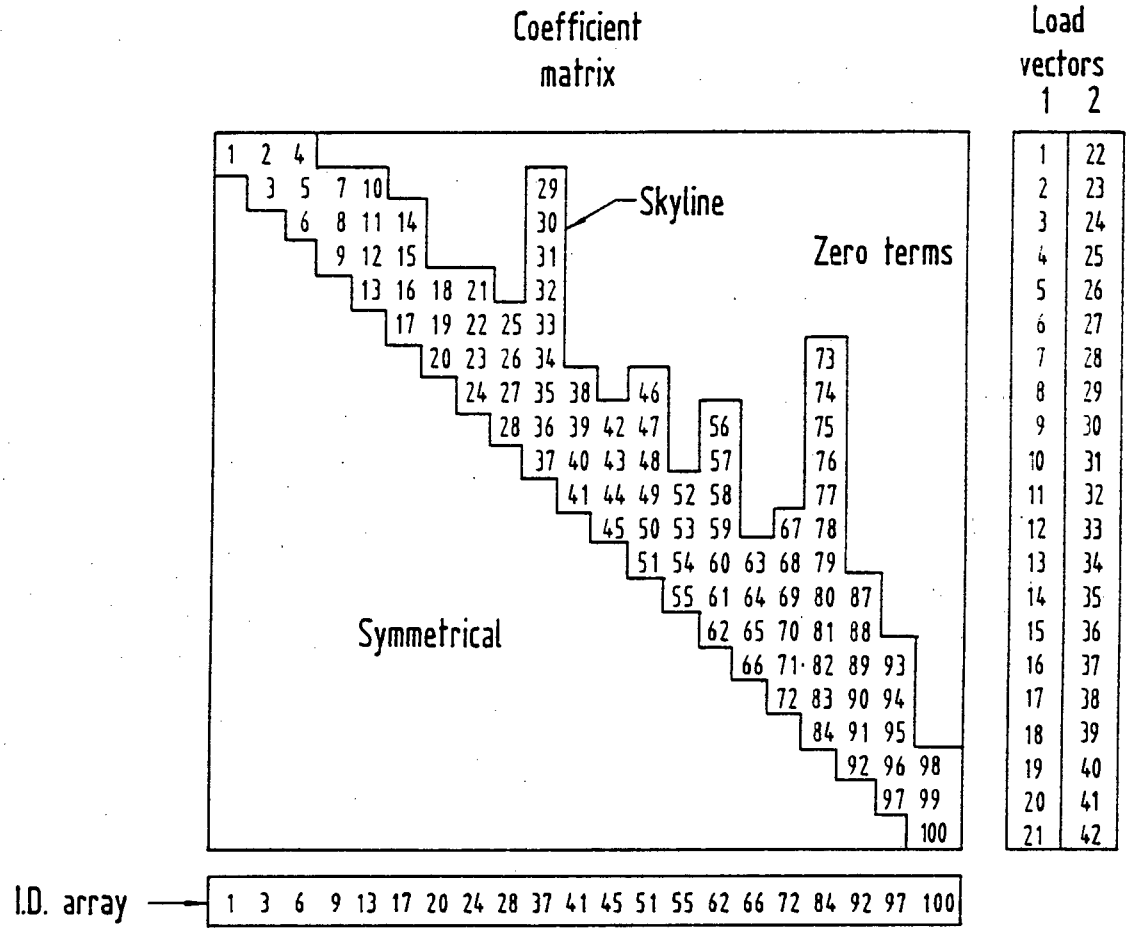


Figure 6.1 : Skyline scheme of storage for the coefficient matrix

6.3 Dynamic v/s block partitioning

This section deals with the module which decomposes the coefficient matrix to the form U^TU .

The solver is based on the well known rigid block partitioned form of solution [24]. The partitions are, however, not rigidly defined, hence the term "dynamic" partitioning. A brief review of the rigid block partitioned form of the algorithm will be given in order to highlight

the salient differences and point out the advantages of dynamic partitioning.

Both rigid block- and dynamic block partitioning depend on the fundamental principle that the decomposition of elements in a particular column of the coefficient matrix only require those (already reduced) terms which fall within the horizontal shadow of the column under consideration (Figure 6.2). [24,30]

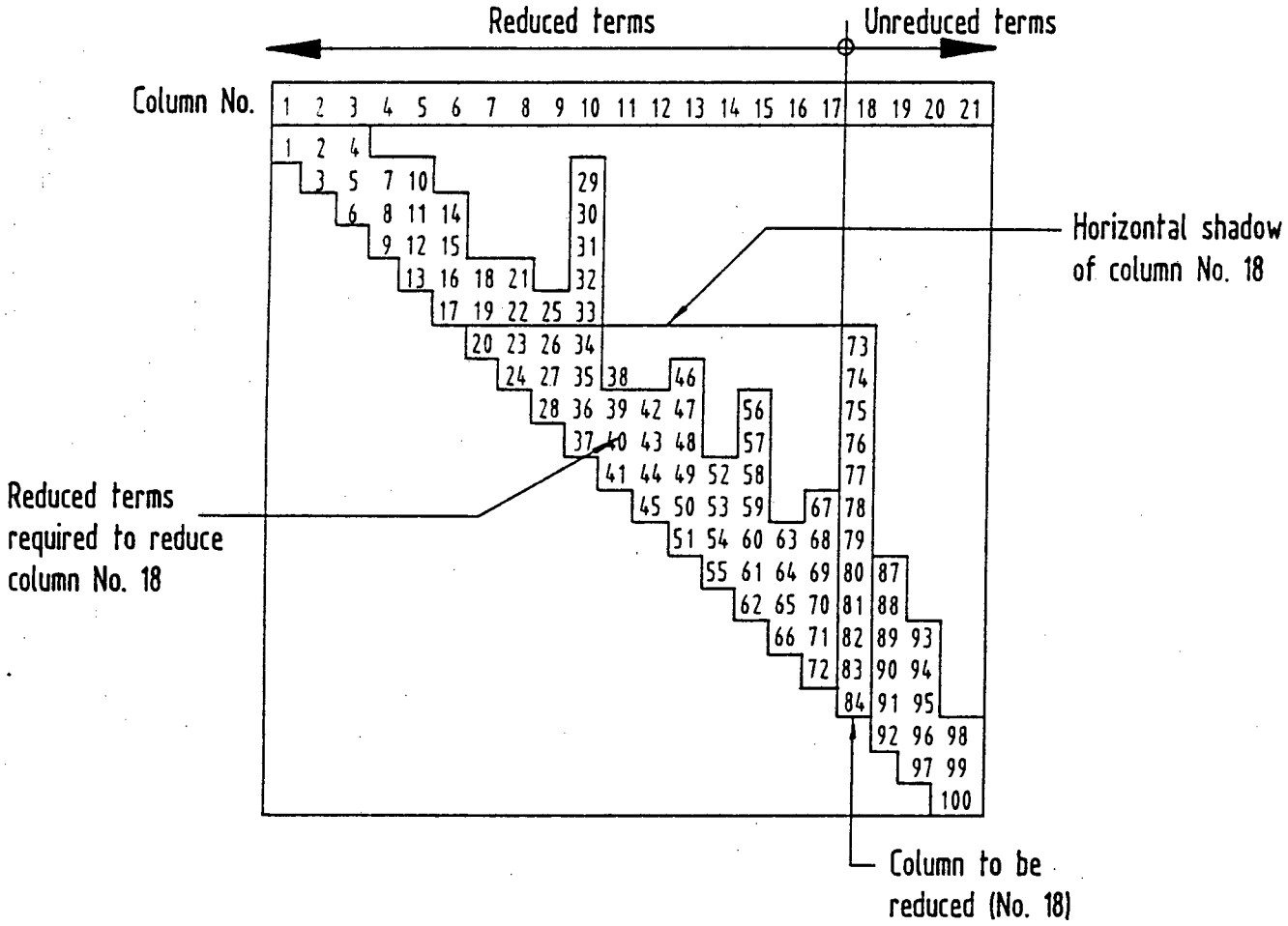


Figure 6.2 : Terms required to reduce a particular column

It is generally faster to work with whole columns at one time. This requires far less house keeping and is more efficient in the transfer of data from auxiliary storage.

With rigid block partitioning, the coefficient matrix is divided into blocks prior to decomposition. The number of terms in each of these blocks is always less than half the number of storage locations available in-core (Figure 6.3).

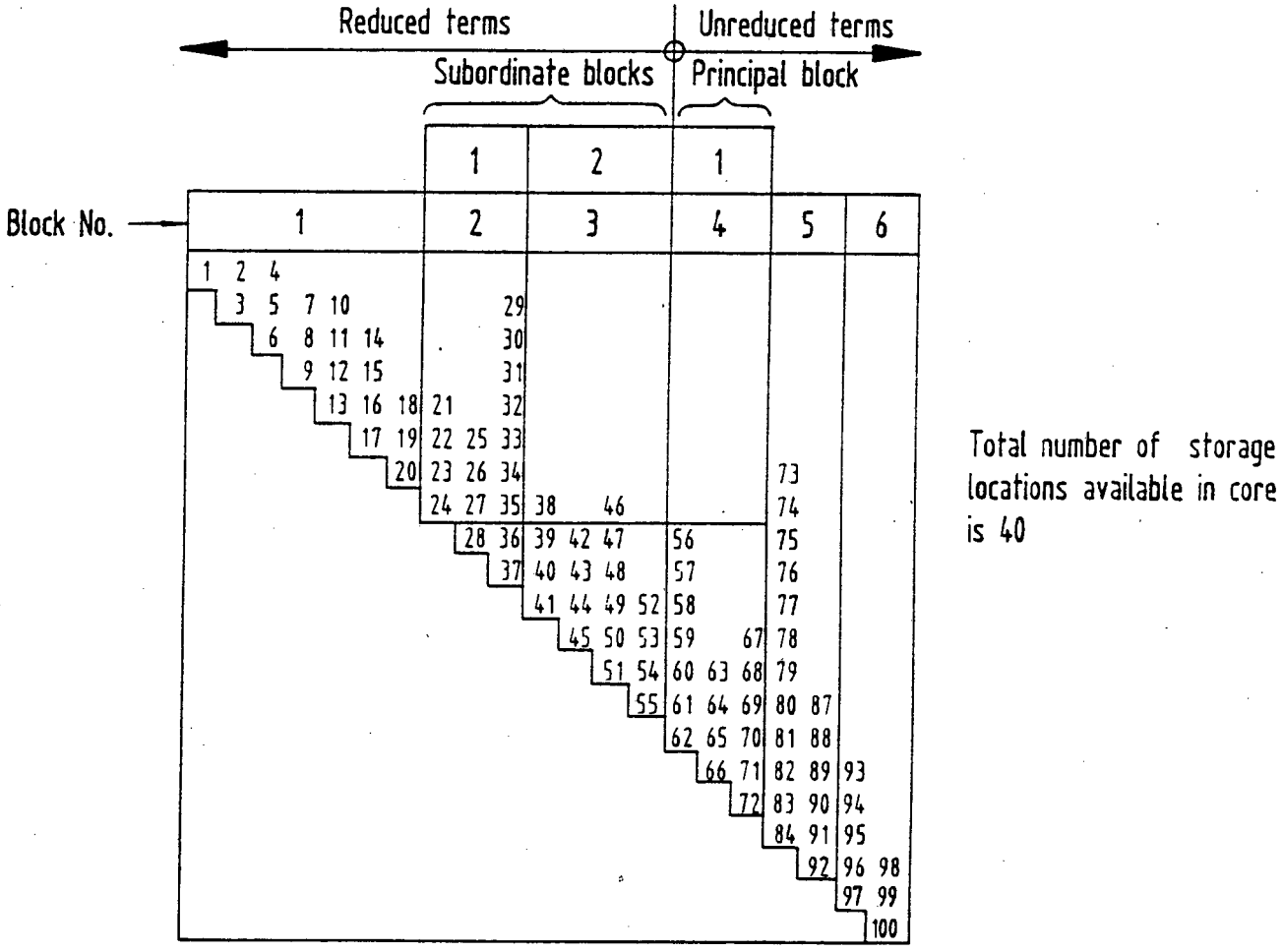


Figure 6.3 : Block partitioning

Having partitioned the coefficient matrix, the solution proceeds as follows :

i) The first two blocks are transferred into core, reduced and the reduced terms transferred to auxiliary storage.

ii) The next block, now termed the principal block, is then loaded into core.

iii) For the principal block, subordinate blocks are defined as those lying wholly or partially within the horizontal shadow of the principal block (Figure 6.3). These subordinate blocks (already reduced) are loaded into core (in ascending order) as the decomposition of the principal block proceeds.

iv) On completion of the decomposition, the principal block is transferred to auxiliary storage.

v) The next un-reduced block now becomes the principal block and is loaded into core. The reduction procedure (ii) to (iv) is repeated.

Dynamic partitioning also approaches the problem in block form. The coefficient matrix is divided into two blocks (principal and subordinate). The smallest block size for either the reduced or unreduced partition is the number of terms in the semi-bandwidth of the coefficient matrix. This ensures that at least one column of each partition can reside in memory at any one time.

The "spread" of any principal block in the coefficient matrix is defined as the sum of the terms in the block and the terms in columns which fall wholly or partially within the horizontal shadow of this block (Figure 6.4). A parameter N defines the minimum size of any one un-reduced block. The selection of this parameter is discussed in a later section. With the paramater N set, partitioning of blocks then proceeds as outlined below.

Presume that the decomposition of the coefficient matrix has advanced some way along the matrix. Un-reduced columns are added sequentially into the principal block until the total number of elements in the block (P) exceeds the minimum block size (N), i.e. : P >= N. The "spread" (S) of this block is then calculated (Figure 6.4).

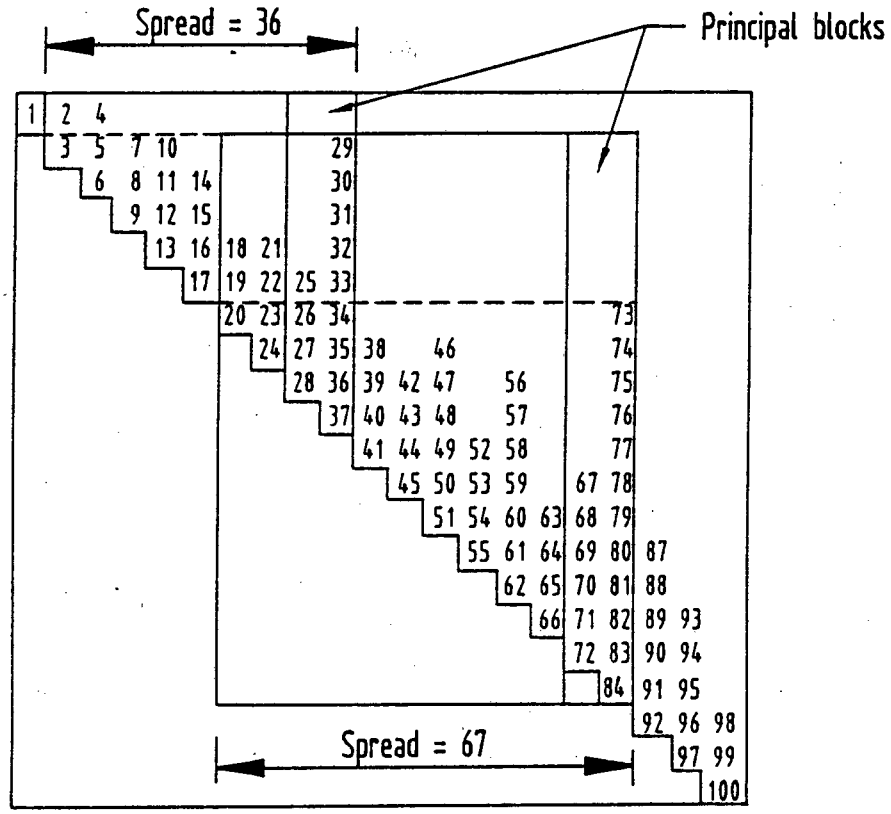


Figure 6.4 : Spread of a principal block

If the total number of available in-core storage locations is M, three possibilities exist, i.e. :

a) S > M

b) S = M

c) S < M

Depending on the value of S the decomposition proceeds as follows :

a) S > M (Figure 6.5(a))

The principal block is transferred to core. Subordinate blocks are partitioned to contain only whole columns, starting with the lowest required decomposed column. The subordinate blocks are sized such that at any one time the principal block and one subordinate block can reside in-core. Decomposition of the principal block then follows along the same lines as in rigid block partitioning.

b) S = M (Figure 6.5(b))

All the elements in the "spread" of the principal block are transferred to core, the principal block is decomposed and transferred to auxiliary storage.

c) S < M (Figure 6.5(c))

The possibility exists that the principal block can be increased in size. The next un-reduced column is added to the principal block and the spread (S) is adjusted. Un-reduced columns are added to the principal block until the addition of one more column would result in : S > M. The decomposition then proceeds along the same lines as in (b).

All the house-keeping and partitioning of principal and subordinate blocks is calculated using the DA array. As only integer arithmetic is involved, this is extremely fast and efficient.

6.4 Parameters

The selection of the parameter N, which determines the minimum size of the principal block, does affect the efficiency of the algorithm. For a matrix with semi-bandwidth of width (SBW) and a machine with (M) available in-core storage locations, the parameter (N) must, of course, lie within the following limits :

$$SBW <= N <= M-SBW \qquad (6.1)$$

In this algorithm N is set as follows (Figure 6.6) :

a) If SBW*SBW >= M/2 then N=M/2

b) If SBW*SBW < M/2 then N=M-SBW*SBW

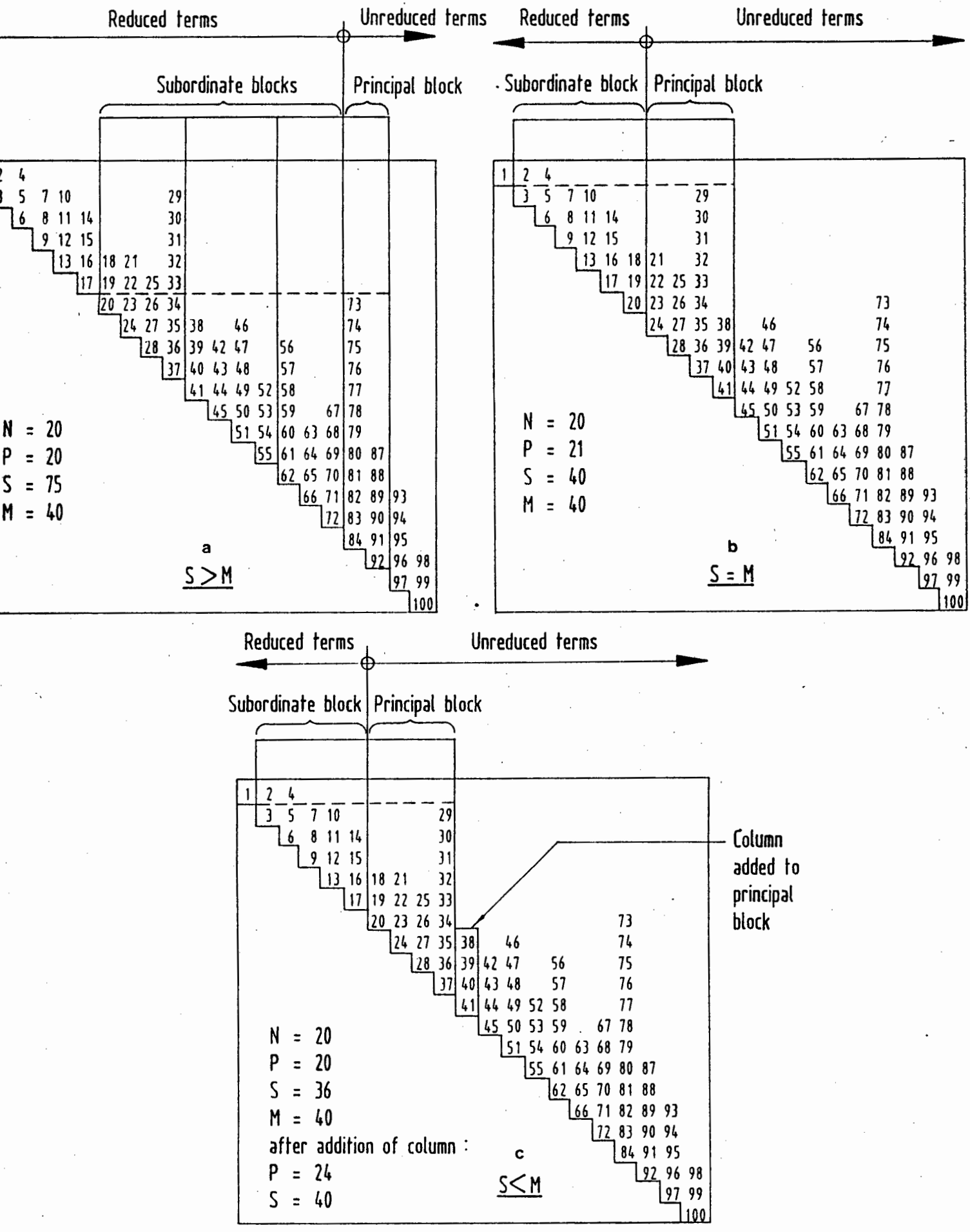


Figure 6.5 : Dynamic partitioning

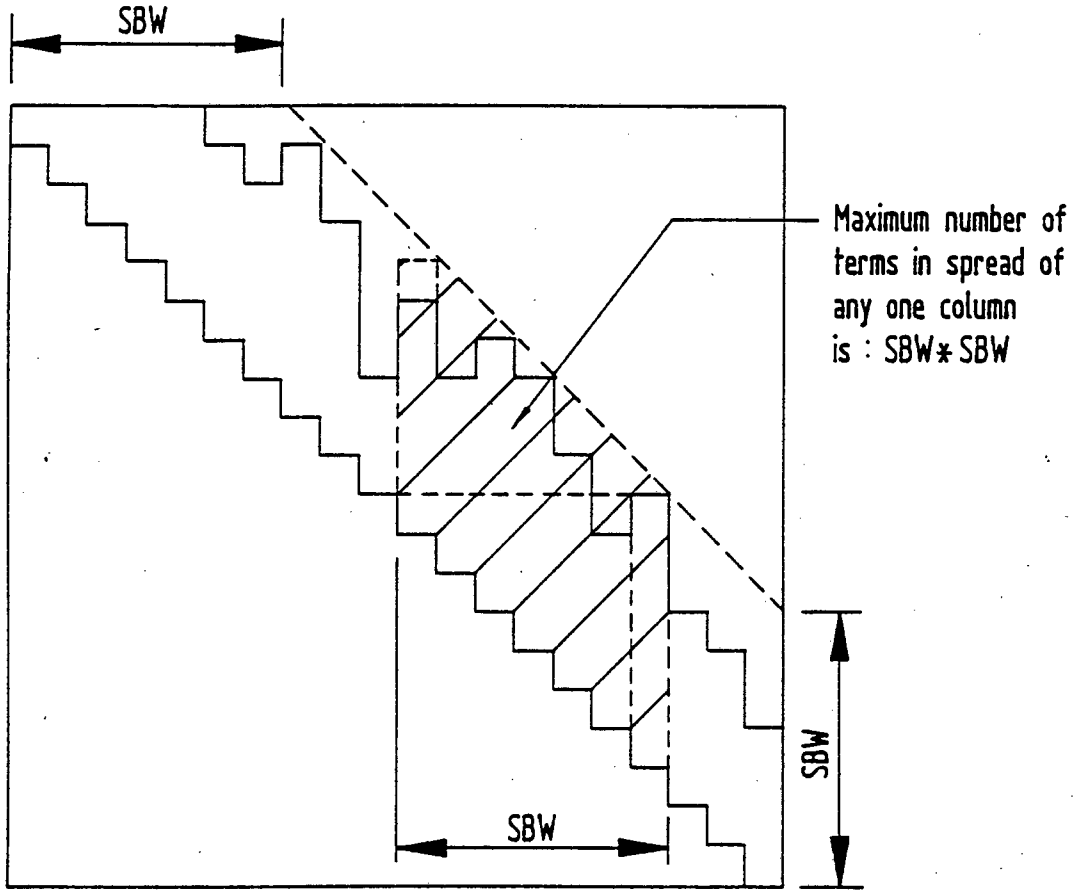


Figure 6.6 : Maximum number of terms in the spread of any column

It is clear that the narrower the bandwidth, the more efficient dynamic partitioning becomes. Rigid block partitioning, on the other hand, becomes progressively less efficient as the bandwidth reduces! (Figure 6.7).

The selection of (N) is by no means an optimum and can, of course, be set to any value within the two extreme limits, (SBW) and (M-SBW).

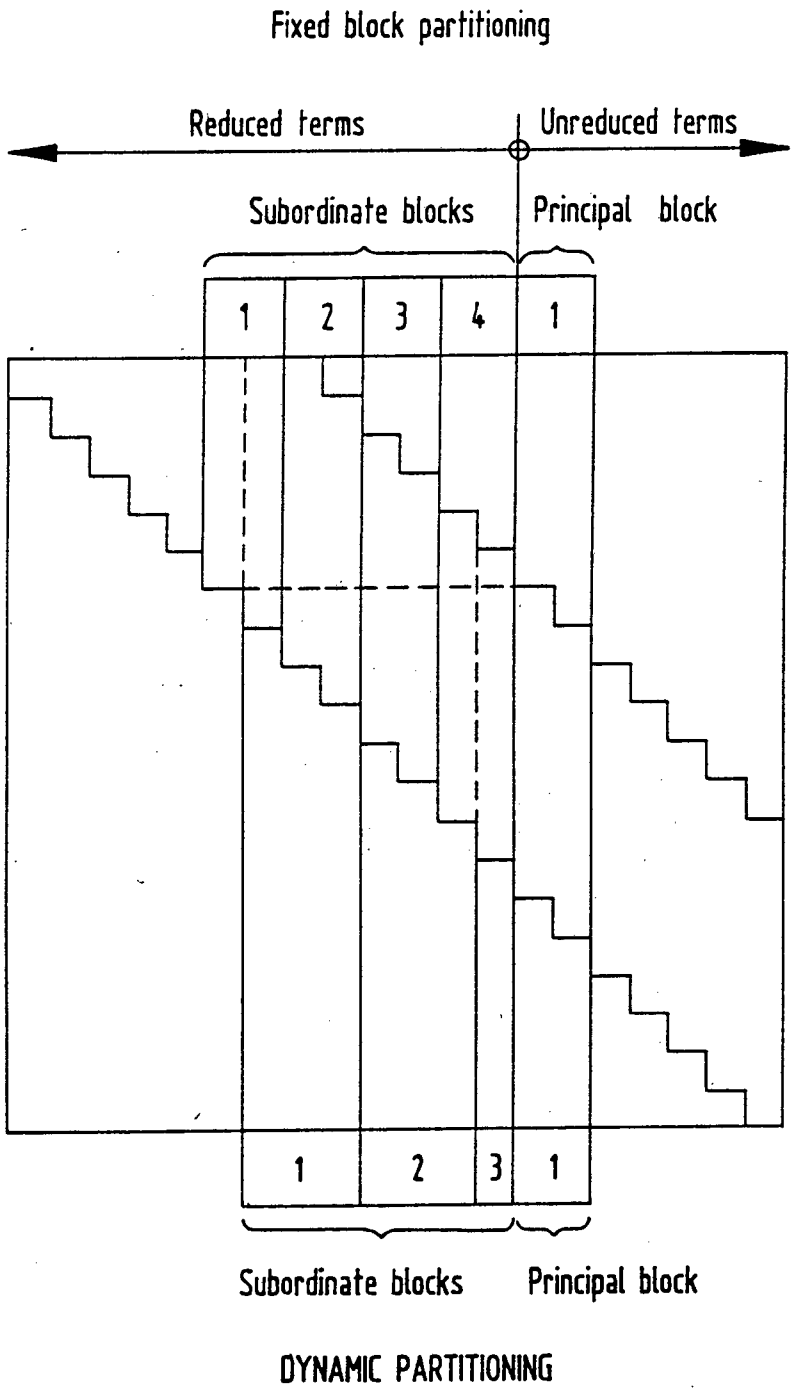


Figure 6.7(a) : Dynamic vs Block partitioning (large bandwidth)

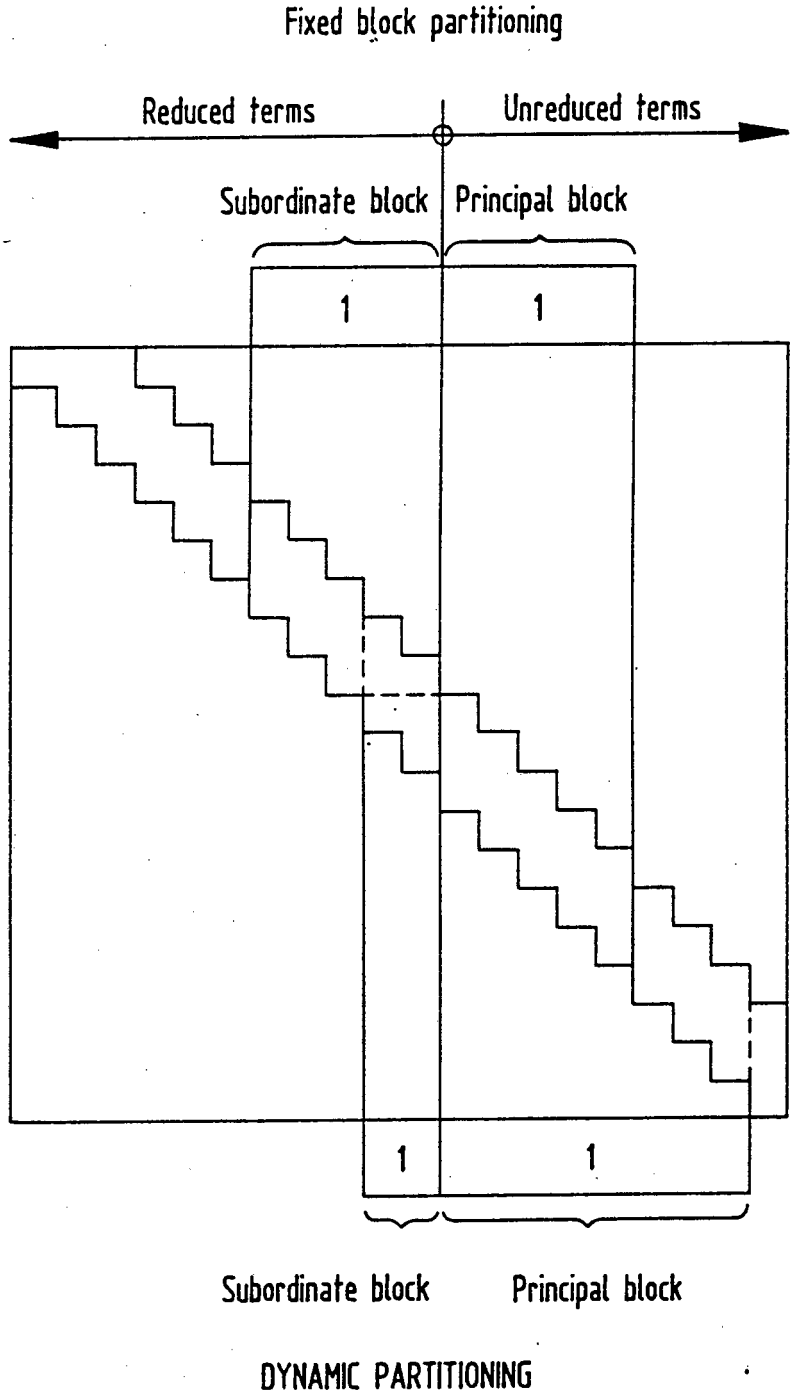


Figure 6.7(b) : Dynamic vs Block partitioning (small bandwidth)

6.5 Forward reduction/back substitution

This section deals with the module which solves the system of linear equations $U^T U\ x = b$. The solution proceeds in two stages, i.e. :

a) Forward reduction : $U^T y = b$ - determine y

b) Back substitution : $U\ x = y$ - determine x

Load vectors are not partitioned. The only limitation on the number of load cases the solver will deal with, is the availability of auxiliary storage. Let the total number of available in-core storage locations be (M) and the total number of terms in the load vector (L).

Assuming that only one load vector will be processed at a time, the total number of in-core storage locations available for the reduced coefficient matrix is then (R = M-L). If the total number of terms in the reduced coefficient matrix is (C), two possibilities exist, i.e. :

a) C > R

b) C <= R

Depending on the value of (R), the solution proceeds as follows :

a) C > R (Figure 6.8(a))

i) Divide the reduced coefficient matrix into blocks. Each block comprises the maximum number of columns that can reside in the remaining in-core space (R).

ii) Transfer the first load vector to be considered to core.

iii) Load the blocks of the reduced coefficient matrix in ascending order into core and then perform a forward reduction.

iv) Load the blocks of the reduced coefficient matrix in descending order into core and perform a back substitution.

v) Transfer the resulting displacement vector to auxiliary storage or print out.

vi) Consider the next load case.

b) C <= R (Figure 6.8(b))

In this instance the entire reduced coefficient matrix plus at least one load vector can reside in-core at any one time.

i) Transfer the entire reduced coefficient matrix into core. These terms will reside in-core for the duration of the solution.

ii) Determine the number of load vectors that can reside in the remaining in-core space ((M-R)/L).

iii) Load as many load vectors into core as possible or until the last load case is encountered.

iv) Perform a forward reduction and a back substitution on these load vectors.

v) Transfer the resulting displacement vectors to auxiliary storage or print out.

vi) Consider the next set of load vectors.

All the house-keeping is done in the main program section using the DA array. The house-keeping is done in integer arithmetic and is extremely efficient.

6.6 Recovering reactions

Reactions at the deleted degrees of freedom can be calculated by using Lagrange multipliers, penalty functions or constraint equations [1]. The method used in this development is based on the latter.

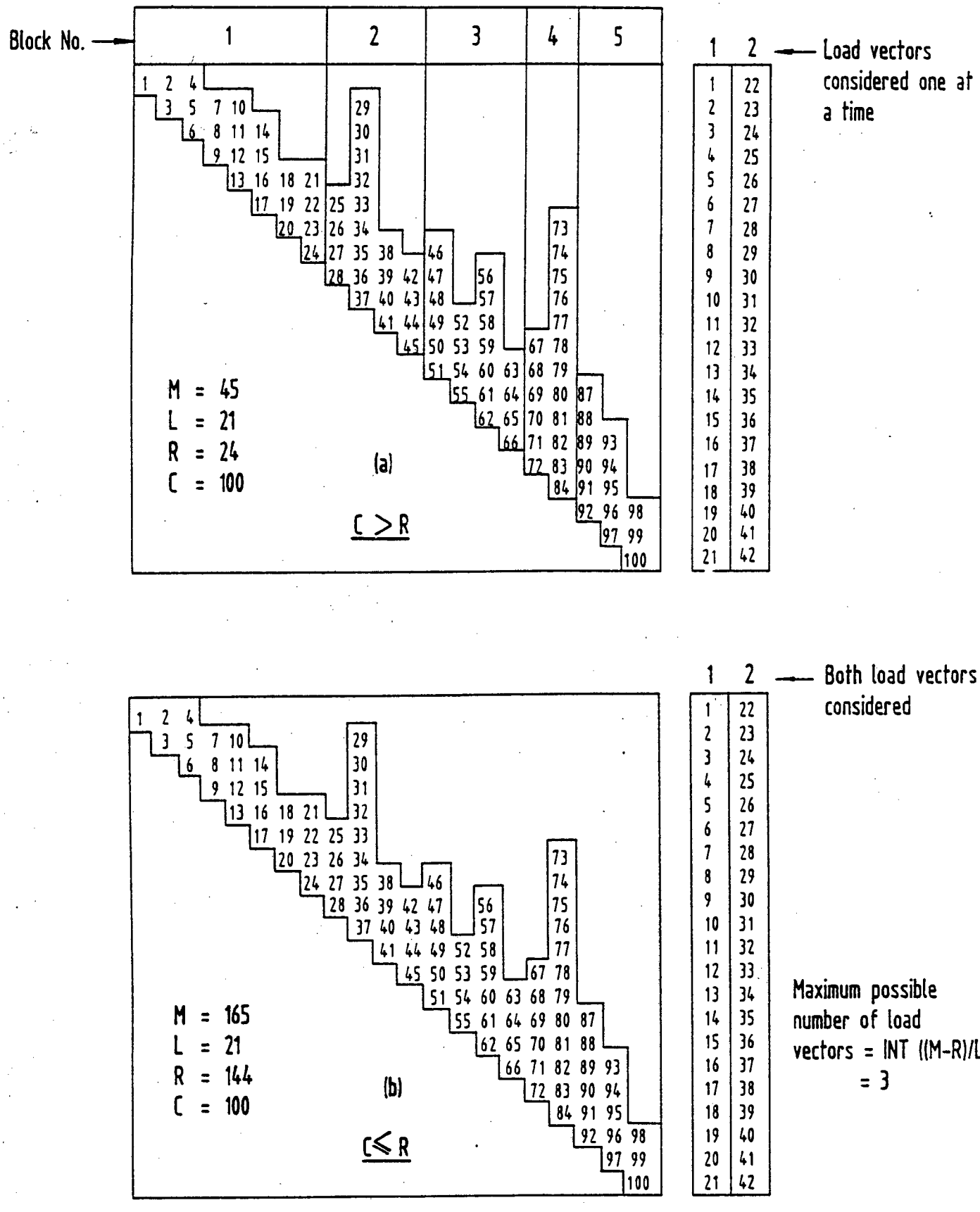


Figure 6.8 : Forward reduction - Back substitution

The structure stiffness matrix is assembled for all degrees of freedom. In order to identify the deleted degrees of freedom, a fixity vector is established. This vector contains one entry for each degree of freedom in the entire model. All constrained degrees of freedom have a corresponding non-zero entry in the fixity vector.

Decomposition of the stiffness matrix then proceeds as described in the previous sections, ignoring the deleted degrees of freedom. While the decomposition proceeds, the constraint equation for each deleted degree of freedom is recovered and stored on auxiliary storage.

Forward reduction and back substitution is also carried out by ignoring the constrained degrees of freedom in the model. After the displacement vector has been established, reactions can be calculated by using the constraint equations on auxiliary storage, i.e. :

$$R = C^T \delta - f \qquad (6.2)$$

where :

- R is the reaction,
- C is the constraint equation (vector),
- δ is the displacement vector and
- f is the loads applied to the degree of freedom.

With some additional programming effort, reactions can be recovered during the back substitution phase of the analysis. This would mean that constraint equations need not be stored on auxiliary storage. Prescribed degrees of freedom can also be accommodated in a similar manner. Elastic supports can be introduced into the program by simply adding the stiffness of the support to the appropriate diagonal term in the stiffness matrix before triangulation.

CHAPTER 7

PRESTRESSING : PROFILES, FORCES AND EQUIVALENT LOADS

7.1 Introduction

The prestressing utility module PROFIL allows the user to :

- Define the profile of a prestressing cable along the length of a member with several spans.

- Calculate the force profile along the length of the cable at various stages of stressing.

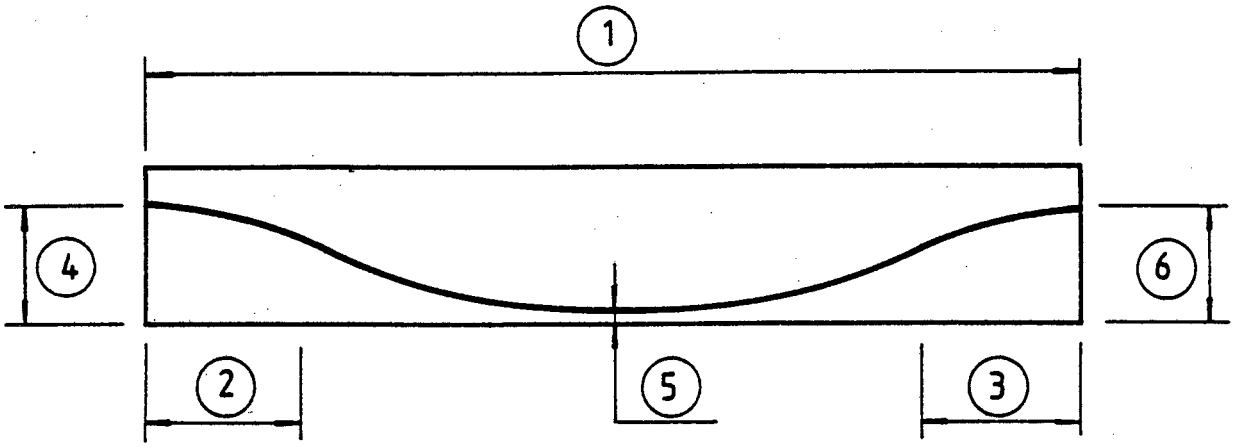
- Calculate the equivalent loading due to the cable profile and stressing force.

Profile data can be stored and retrieved from disk. An option allows the data to be printed. Numerous profiles can be examined and amended very quickly and easily.

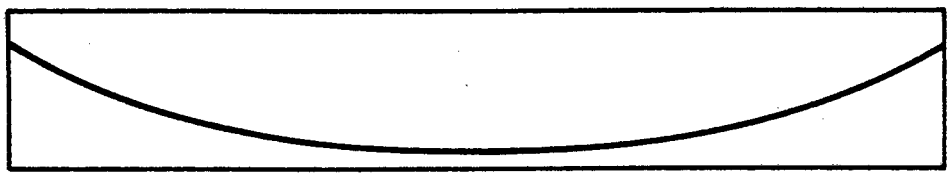
7.2 Cable profiles

Prestressing cables can be installed to virtually any profile, the most common of which are straight and parabolic. Other profiles (cubic parabola, sine curves etc) are used in unusual instances where specific variations in loading must be accommodated. The profile module was built to accommodate only straight and parabolic cables.

Figure 7.1(a) shows the six dimensions required to define the basic cable profile in any one "span". By setting some of these dimensions to zero, six other profiles can be defined (Figure 7.1(b)-(g)). Spans can be combined to produce the profile along the total length of the cable. The term "span" is not used here in the customary sense, but serves to indicate any one of the seven possible profiles shown in Figure 7.1.



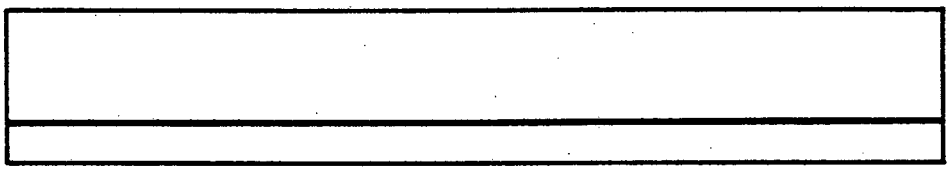
(a) Profile type 1



(b) Profile type 2

$$(3) = (2) = 0$$

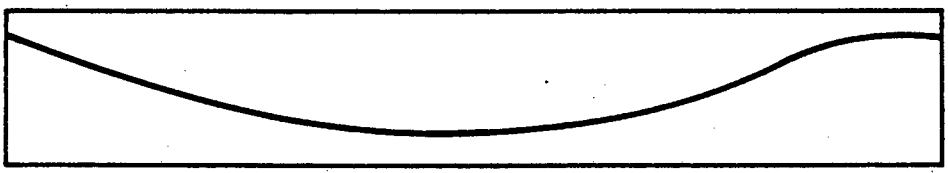
$$(4) > (5) \quad (6) > (5)$$



(c) Profile type 3

$$(3) = (2) = 0$$

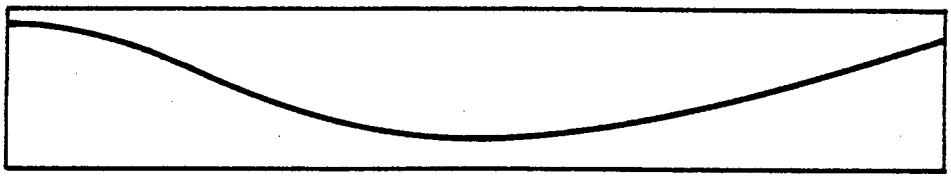
$$(4) = (5) = (6)$$



(d) Profile type 4

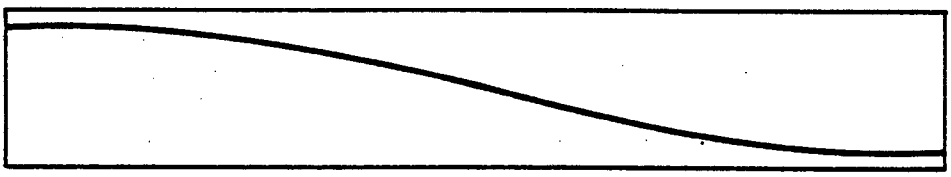
$$(2) = 0$$

Figure 7.1(a-d) : Basic cable profiles 1-4



(e) Profile type 5

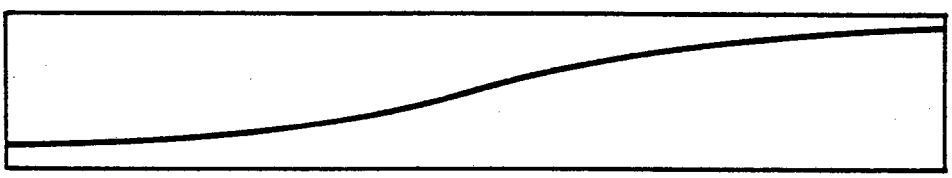
(3) = 0



(f) Profile type 6

(3) = 0

(5) = (6)



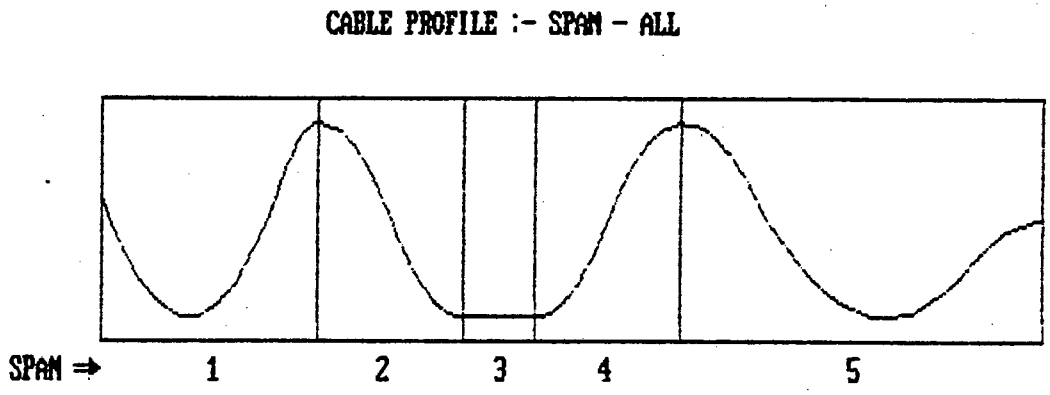
(g) Profile type 7

(2) = 0

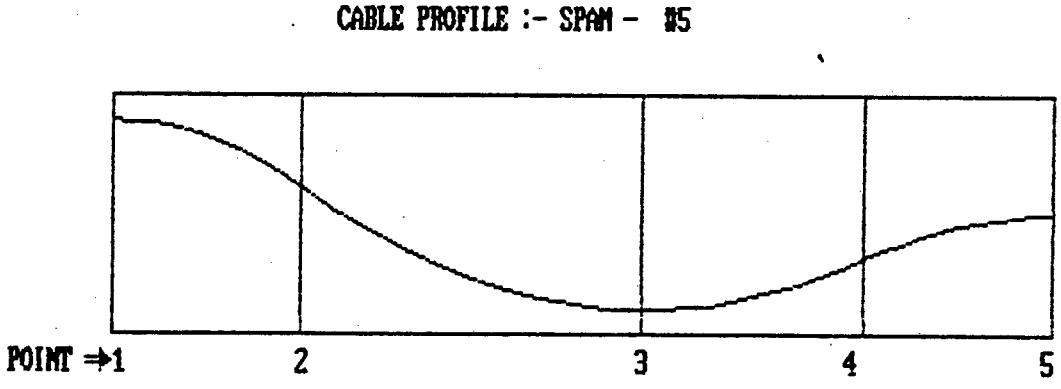
(4) = (5)

Figure 7.1(e-g) : Basic cable profiles 5-7

The program checks the consistency of the data and compatibility of dimensions. Errors or conflicting data are reported. Profiles can be viewed for the whole length of the cable (Figure 7.2(a)) or for selected "spans" (Figure 7.2(b)). Control point values as well as the height of the cable centroid at 1/10th points along the "span" are displayed (Figure 7.2(b)).



(a) ALL SPANS



POINT	X	Y	1/10 POINT	X	Y	1/10 POINT	X	Y
1	8.000	0.900	0.0	8.000	0.900	0.5	10.500	0.116
2	9.000	0.616	0.1	8.500	0.829	0.6	11.000	0.105
3	10.821	0.100	0.2	9.000	0.616	0.7	11.500	0.172
4	12.000	0.316	0.3	9.500	0.372	0.8	12.000	0.316
5	13.000	0.500	0.4	10.000	0.205	0.9	12.500	0.454

(b) ONE SPAN

Figure 7.2 : Cable profiles

7.3 Force profiles

Once the cable profile has been established, the force profile can be calculated. The force profile is a graphic representation of the variation in the prestressing force, due to losses, along the length of the cable. In depth discussions of the losses in prestressed concrete can be found in most good references [35,36]. PROFIL includes the following loss calculations :

7.3.1 Friction losses

In post-tensioning systems, friction on the cable causes a progressive reduction in prestressing force as the distance from the jacking end increases. Friction on the cable is due to unintended curvature resulting from the placement of the system and the intended curvature of the cable. Friction losses are calculated from the following well known formula [35] :

$$F_x = F_o \, e^{(-kx-\mu\theta)} \qquad (7.1)$$

where :

- F_x is the force at a distance x from the jacking end,
- F_o is the jacking force,
- k is the so called "wobble" coefficient,
- μ is the curvature coefficient and
- θ is the total angular rotation of the cable, from the jacking end up to the point under consideration.

7.3.2 Wedge set

In post-tensioning systems, the cable is "locked off" at the jack after stressing. This lock off always results in some slipping of the cable at the anchorage and is termed "wedge set". As the cable slips back, friction forces act on the cable in a direction opposed to that during stressing. Some loss of prestressing force at the jack is inevitable.

The distance which the wedge slips back at lock off depends on the stressing system used. The length over which this wedge set affects the force profile, and hence the profile itself, can be calculated from the following relationship :

$$dL = \int_0^{\ell} \frac{dF dx}{E_s A_s} \qquad (7.2)$$

where :

- dL is the wedge set,
- ℓ is the length of the cable affected by the wedge set,
- dF is the change in the prestressing force over an infinitesimal distance dx ,
- A_s is the cross-sectional area of the cable and
- E_s is the elastic modulus of the cable.

It is clear that the wedge set is proportional to the hatched area in Figure 7.3(a).

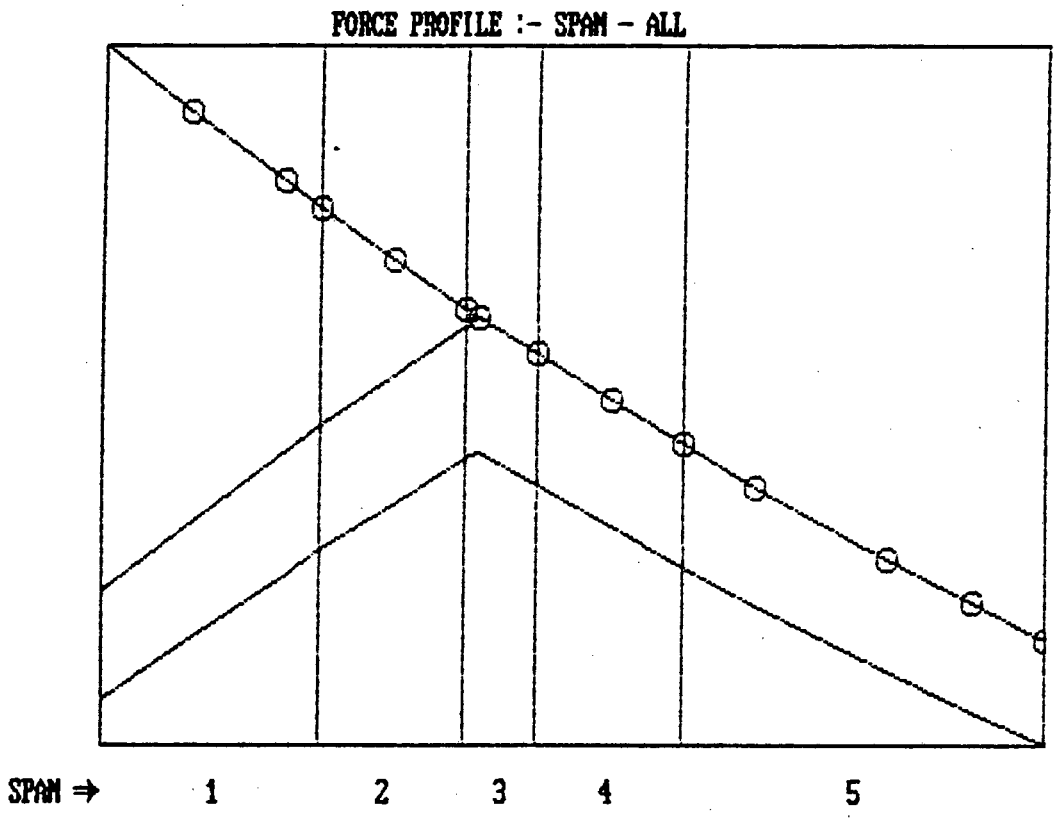
7.3.3 Long term losses

Long term losses include the creep and shrinkage in the concrete and relaxation of steel. These losses are usually expressed as a percentage of the prestressing force in the cable.

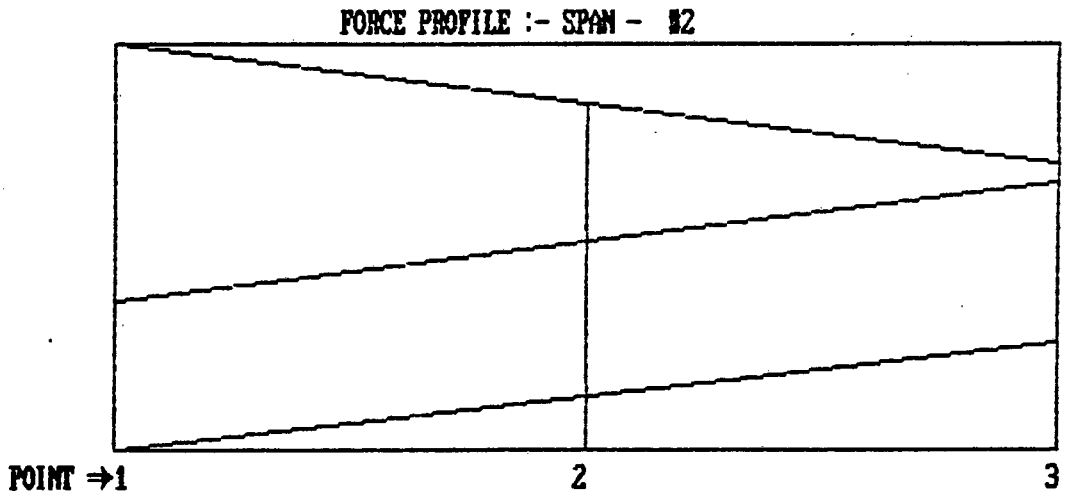
Plots include the forces at jacking, after wedge set and after long term losses. The force profile can be viewed for the whole length of the cable (Figure 7.3(a)) or for selected spans (Figure 7.3(b)). Control point values are displayed (Figure 7.3(b)). Stressing can occur from either or both ends of the cable.

7.4 Equivalent loads

The "equivalent load" concept is a technique whereby the effects of a prestressing cable on any structural member is replaced by a system of external, statically equivalent loads [37,38].



(a) ALL SPANS

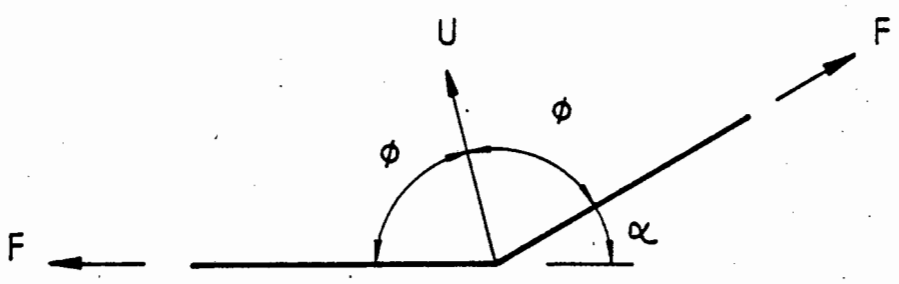


POINT	X-DIST.	JACKING	AFTER W/S	LONG TERM
1	3.0000	116.98	101.15	92.847
2	4.0000	113.25	104.88	95.441
3	5.0000	109.64	108.49	98.726

(b) ONE SPAN

Figure 7.3 : Force profile

Each change in direction of the prestressing cable gives rise to radial forces. If friction forces are ignored, the resultant radial force at the change in direction is directed along the bisector of the angle (Figure 7.4). In the normal prestressing systems, the deflection angles are small and no significant errors will be made if the radial forces are assumed to act vertically. The equivalent loads due to a parabolic section of cable can be assumed to be uniformly distributed.

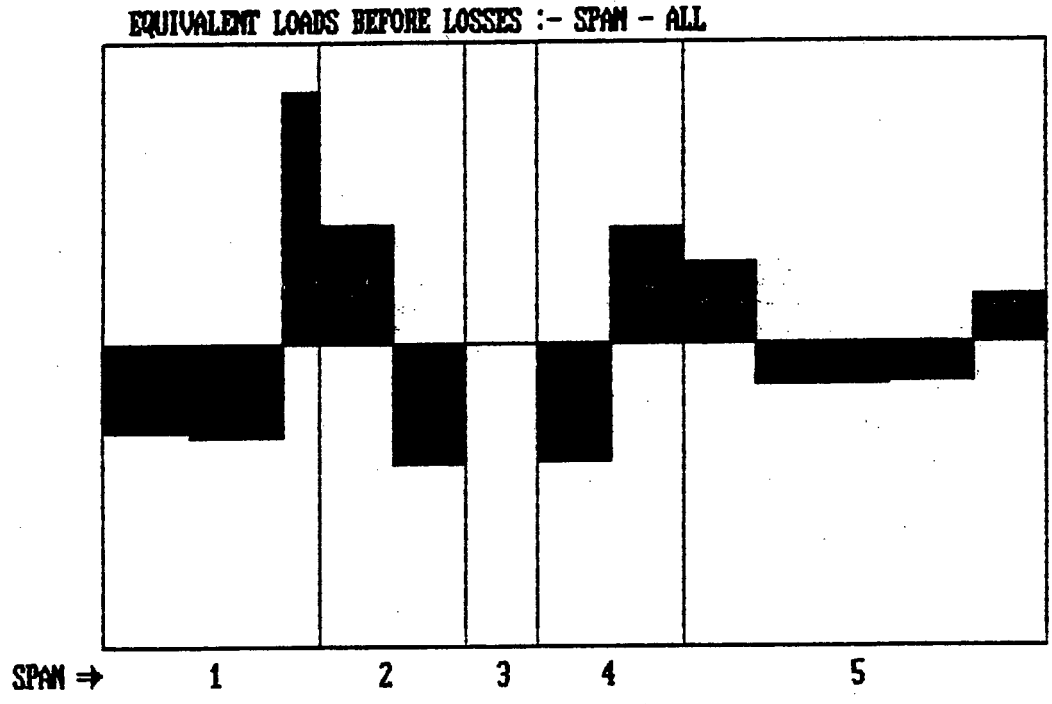


F = Prestressing force
U = Radial force due to change in cable direction

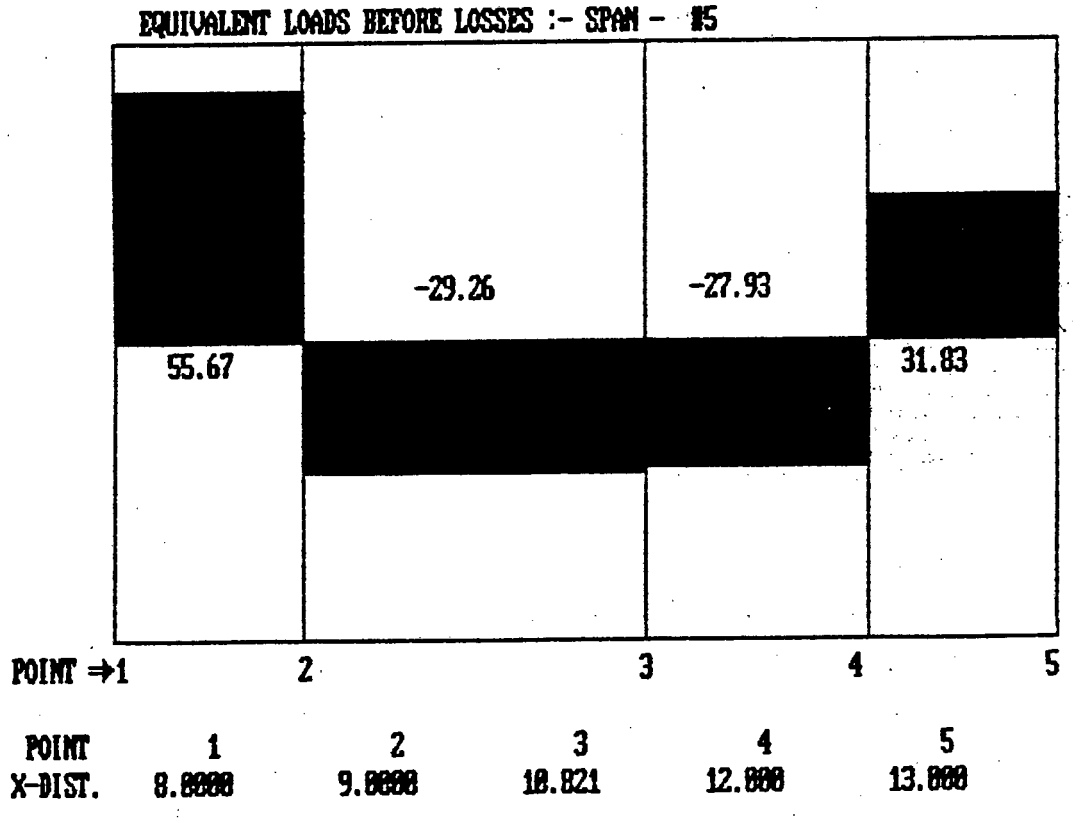
Figure 7.4 : Radial forces due to prestressing

For each parabolic section of cable, the program assumes a constant cable force equal to the average force in that section. The equivalent loads are then uniformly distributed. These loads can be viewed for the whole length of the cable (Figure 7.5(a)) or for selected "spans" (Figure 7.5(b)).

The system of externally applied transverse and in-plane equivalent loads can reproduce the state of stress in the structural system due to the prestressing cable exactly. Elastic behaviour of the structural system under these equivalent loads can be determined by a finite element analysis of the structure.



(a) ALL SPANS



(b) ONE SPAN

Figure 7.5 : Equivalent loads

No interface is supplied between the profile module and the load case module of the program. The user is, therefore, free to apply the equivalent loads due to prestressing as line- or area loads depending on the specific problem under consideration.

CHAPTER 8

DESIGN OF PRESTRESSED FLAT SLABS

8.1 Introduction

Structural design is still very much an art that must be acquired through knowledge and experience. The objective of the current development is to provide the design engineer with useful practical tools to aid in the structural design of prestressed flat slabs.

In structural engineering, the distinction between theory, analysis and design is usually not clearly defined. With elements such as prestressed concrete and especially prestressed flat slabs, the distinction becomes even less clear.

The design dictates the analysis and much "theory" is not theory in the accepted sense. The two criteria of serviceability and failure, for instance, require totally different approaches to theory, analysis and design.

Aspects of the theory and analysis of prestressed flat slabs have been discussed in the preceding chapters. The design tools required to carry out successful design are discussed in this chapter.

Design tools provided in "PRESLAB" are by no means exhaustive, but the elastic behaviour of the model is adequately described and the information available on auxiliary storage. With the modular approach to the design of the program, additional structural design modules can very easily be built and incorporated.

8.2 Design of prestressed concrete flat slabs

Several criteria for the behaviour of prestressed concrete elements can be considered essential or desirable. The loading history of a prestressed concrete element as presented by Lin [36] illustrates this.

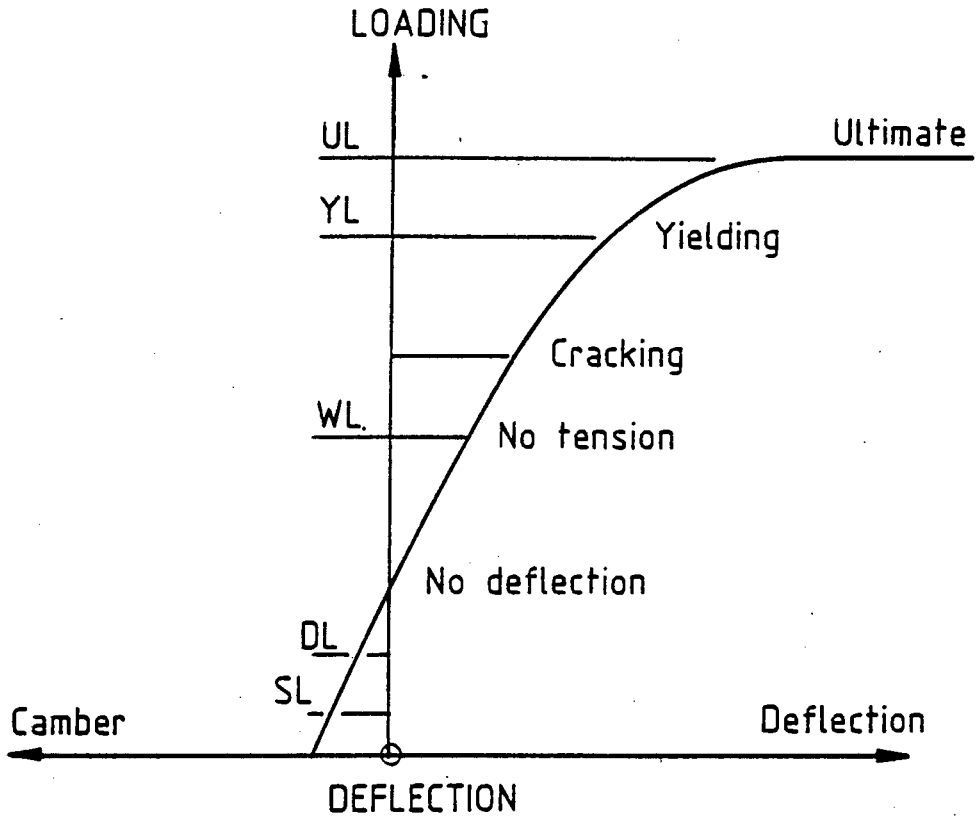


Figure 8.1 : Loading history of a prestressed element in bending

Five points in the behaviour of the structure can be identified, i.e. :

a) No deflection - corresponding to a rectangular stress block (i.e. the plate is in plane stress)

b) No tension - triangular stress block

c) Point of cracking - modulus of rupture

d) Onset of yield of the steel

e) Ultimate strength - failure of the structure.

Five loading conditions can be identified, i.e. :

a) SL - Self weight of the structure

b) DL - Total dead load

c) WL - Working load (= dead load + imposed load)

d) YL - Yield load (= k1 x WL) at which yield of the steel occurs

e) UL - Ultimate load (= k2 x WL) at which the structural element fails.

Most commonly, the two criteria of analysis and design are :

- matching the working load (WL) and the condition of no tension in the case of "stress" design and

- matching ultimate load (UL) and ultimate strength in the case of "strength" design.

An examination of Figure 8.1 shows that other criteria are possible. In the case of indeterminate prestressed concrete structures, no deflection offers an alternative point of departure. From a serviceability point of view, no cracking is a further design consideration.

Choosing the design criterion is fundamental to prestressed concrete design and poor judgement at this stage can have disastrous consequences. Limit state design attempts to address all necessary considerations in a single philosophy. Ultimately though, one or other of the criteria must be chosen as the critical one and the design carried out for that criteria, due consideration being given to "satisfactory behaviour" in all other aspects.

8.3 Load balancing

As pointed out in Chapter 7, any change in the alignment of a prestressing cable produces a lateral force in the structure at that point. The distribution and magnitude of the transverse loads depend on the profile and the prestressing force applied to the cable respectively.

These transverse loads can be viewed as the static equivalent of the prestressing effects, i.e. : if these loads are applied as external loads to the structure, the structure will deform in exactly the same manner as if the prestressing had been applied directly to the structure. Transverse loads due to prestressing are termed "equivalent" loads and are fundamental to the load balancing concept.

The state of no transverse deflection in Figure 8.1 indicates the point where the slab is in plane stress. This condition is achieved by "balancing" the dead load and a proportion of the imposed load with equivalent transverse loads due to prestressing.

The proportion of imposed load included in the total load to be balanced is very much problem dependent. No deflection during the major portion of the life of the structure is normally a desirable criteria.

8.4 Design procedure

A suggested design procedure is as follows [43] :

a) Analyse the slab under self weight, permanent applied load and imposed loads.

b) Using these results, propose a cable layout, profile and force.

c) Analyse the slab under prestress using equivalent loads.

d) View the deflected shape of the slab under various factored load combinations (dead load, imposed loads and prestress loads).

e) Modify the cable layout, profile or force, if necessary. Repeat steps 3 and 4 until the slab displays no significant deflection under the load combinations.

f) Design untensioned reinforcement to resist the resulting tensile stresses at ultimate loads. Keep in mind that the load factor for prestressing at ultimate loads is 1.0.

g) Carry out ultimate load check for punching shear failure.

h) Calculate the principal stresses in the top and bottom fibres of the slab at the service load. This will indicate the state of compressive stress in the slab and if cracking has occurred.

i) Calculate crack widths.

j) Amend the design if the crack widths are excessive.

k) Check shear stresses.

It is interesting to note how the deflection plot (PLOTDEF) and load combination (COMBI) modules (cf. Chapter 4) can be employed to assess the deflections of the structure under various loading conditions. With the aid of these modules and the equivalent load module (PROFI), prestressing cable profiles and layouts can be determined very quickly and easily for the most complicated slabs and loading conditions.

As there is an infinite number of solutions to the load balancing problem in two dimensions, optimization can now be attempted with greater ease, speed and confidence.

8.5 Design of untensioned reinforcement

Prestressed concrete slabs are, by nature, statically indeterminate and states of no tension are difficult, if not impossible, to achieve throughout the slab. Partially prestressed concrete offers a means of approaching the design of these structures.

In this approach, prestressing is proportioned to satisfy a limit state other than that of ultimate loads. Limit states of deflection and cracking are normally used. Untensioned steel is then added to satisfy the ultimate load limit state. These structures are termed "class 3" in SABS 0100 [50] and CP 110 [49] codes of practice.

Partial prestressing allows a logical mix of prestressing and untensioned steel to satisfy the ultimate limit state and often produces more economical designs. Provision of untensioned steel also controls the formation of cracks in regions of high tensile stress.

A number of methods have been developed for proportioning the reinforcing steel of concrete slabs subjected to a combination of stress resultants. These methods are all based on plasticity theory.

Nielsen [39] developed equations for slab elements in plane stress. Wood [40] and Armer [41] derive equations for slab elements subjected to a triad of bending moments. Morley [42] introduces a fictitious rate of energy dissipation to account for moments and in-plane forces simultaneously. This is shown to be a generalization of the previous equations. Clark and West [43] give some very practical advice on the application of these equations.

The method proposed by Morley [42], although attractive, is very cumbersome. He suggests that, as an alternative, the applied moments and membrane forces can be resolved into two systems of in-plane forces at the level of the top- and bottom steel respectively. Nielsen's [39] equations can then be applied to each system separately. This is also the approach suggested by Clark [48] in his handbook to BS 5400.

In the absence of clearly defined design rules for prestressed flat slabs, the design module offers two alternatives to aid the

designer, i.e. :

- Wood/Armer moments are printed for each element in any two specified directions. Equivalent in-plane forces are also presented in the specified directions.

- The area of untensioned steel required in any two reinforcing directions for each element is printed. This steel area is based on an ultimate load analysis as suggested by Clark [43].

8.5.1 Wood/Armer moments

Wood [40] derived general formula for the required moments of resistance in a flat slab subjected to any moment triad. His formula are only valid for orthogonal reinforcement. Armer [41] extended these formula to include the general case of skew reinforcement. The derivation of these formula is extensively described in the literature and will not be repeated here.

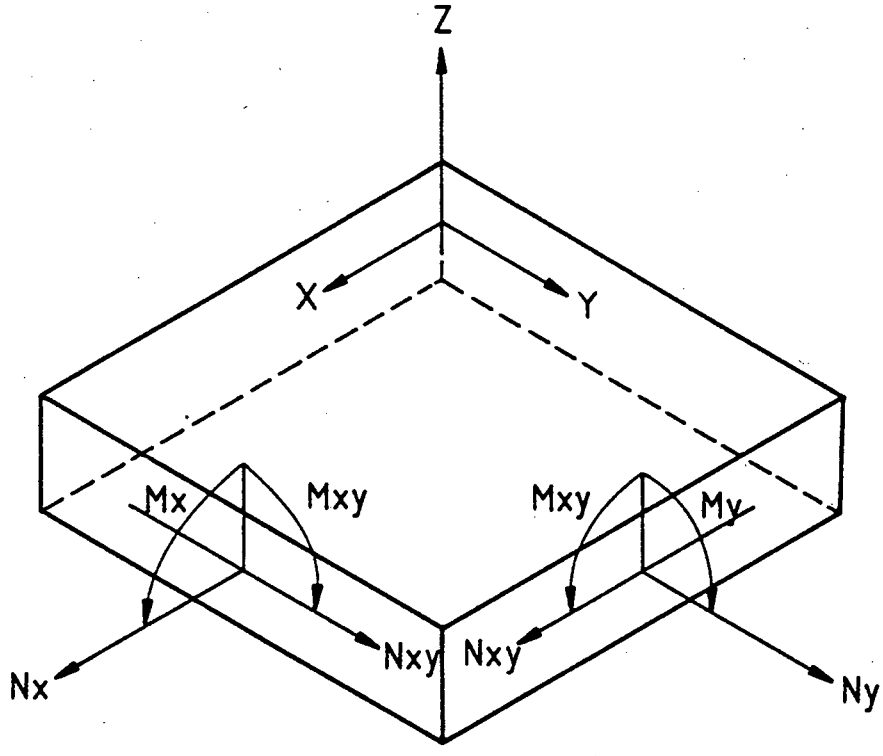


Figure 8.2 : Sign convention for moments and in-plane forces

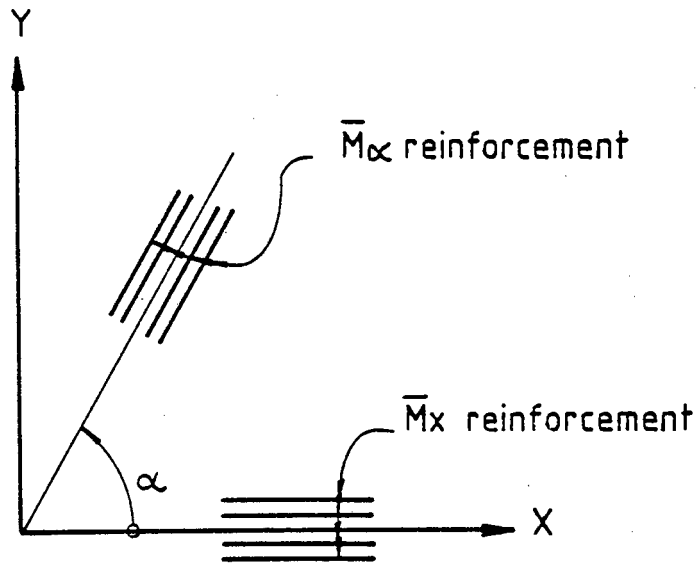


Figure 8.3 : Positive directions of skew reinforcement

For the moments and in-plane forces defined in Figure 8.2 and the system of skew reinforcement defined in Figure 8.3, the required moments of resistance and equivalent in-plane forces can be calculated from :

a) Moments of resistance :

Bottom steel :

$$\bar{M}_x = - M_x + 2 M_{xy} \cot\alpha - M_y \cot^2\alpha + \left| \frac{M_{xy} - M_y \cot\alpha}{\sin\alpha} \right| \qquad (8.1)$$

$$\bar{M}_\alpha = - M_y \operatorname{cosec}^2\alpha + \left| \frac{M_{xy} - M_y \cot\alpha}{\sin\alpha} \right| \qquad (8.2)$$

If $\bar{M}_x < 0$, then $\bar{M}_x = 0$, and

$$\bar{M}_\alpha = \operatorname{cosec}^2\alpha \left\{ - M_y + \left| \frac{(M_{xy} - M_y\cot\alpha)^2}{(M_x - 2M_{xy}\cot\alpha + M_y\cot^2\alpha)} \right| \right\} \qquad (8.3)$$

If $\bar{M}_\alpha < 0$, then $\bar{M}_\alpha = 0$, and

$$\bar{M}_x = - M_x + 2\, M_{xy}\cot\alpha - M_y\cot^2\alpha + \left| \frac{(M_{xy} - M_y\cot\alpha)^2}{M_y} \right| \qquad (8.4)$$

If $\bar{M}_x < 0$ and $\bar{M}_\alpha < 0$, then $\bar{M}_x = 0$, and $\bar{M}_\alpha = 0$.

Top steel :

$$\bar{M}_x = M_x - 2\, M_{xy}\cot\alpha + M_y\cot^2\alpha + \left| \frac{M_{xy} + M_y\cot\alpha}{\sin\alpha} \right| \qquad (8.5)$$

$$\bar{M}_\alpha = M_y\operatorname{cosec}^2\alpha + \left| \frac{M_{xy} - M_y\cot\alpha}{\sin\alpha} \right| \qquad (8.6)$$

If $\bar{M}_x < 0$, then $\bar{M}_x = 0$, and

$$\bar{M}_\alpha = \operatorname{cosec}^2\alpha \left\{ M_y + \left| \frac{(M_{xy} - M_y\cot\alpha)^2}{(M_x - 2M_{xy}\cot\alpha + M_y\cot^2\alpha)} \right| \right\} \qquad (8.7)$$

If $\bar{M}_\alpha < 0$, then $\bar{M}_\alpha = 0$, and

$$\bar{M}_x = M_x - 2\, M_{xy}\cot\alpha + M_y\cot^2\alpha + \left| \frac{(M_{xy} - M_y\cot\alpha)^2}{M_y} \right| \qquad (8.8)$$

If $\bar{M}_x < 0$ and $\bar{M}_\alpha < 0$, then $\bar{M}_x = 0$, and $\bar{M}_\alpha = 0$.

b) In-plane forces :

$$\bar{N}_x = N_x - 2\,N_{xy}\cot\alpha + N_y\cot^2\alpha + \left| \frac{N_{xy} - N_y\cot^2\alpha}{\sin\alpha} \right| \qquad (8.9)$$

$$\bar{N}_\alpha = N_y\,\mathrm{cosec}^2\alpha + \left| \frac{N_{xy} - N_y\cot\alpha}{\sin\alpha} \right| \qquad (8.10)$$

Untensioned steel can be designed using the modular ratio theory in each steel direction. If no in-plane forces exist, the moment transformations can be carried out on the elastic moments resulting from the applied ultimate loads. Design of the untensioned steel can then be carried out using ultimate limit state theory. Top and bottom reinforcement must be designed for separately.

In the derivation of these formula it is assumed that the concrete is capable of carrying the applied compressive stresses. This approach over-estimates the steel areas and under-estimates the concrete stresses. Conservative steel areas are not catastrophic, but a potential compressive failure of the concrete is cause for alarm.

Clark and West [43] suggest that the concrete strength should be at least three times greater than the stress due to the greatest minor principal moment calculated at the service load. To assist the designer in his assessment of the concrete stresses, principal stress results are included in the design module (cf. section 8.6).

8.5.2 Ultimate load design of untensioned reinforcement

Marti and Kong [44] show that for slabs under pure torsion moments, the principal strains in the interior of the slab are always tensile. If it is assumed that concrete cannot carry tensile stresses, the moments and in-plane forces in the slab will be resisted primarily by the regions close to the top and bottom surfaces of the slab.

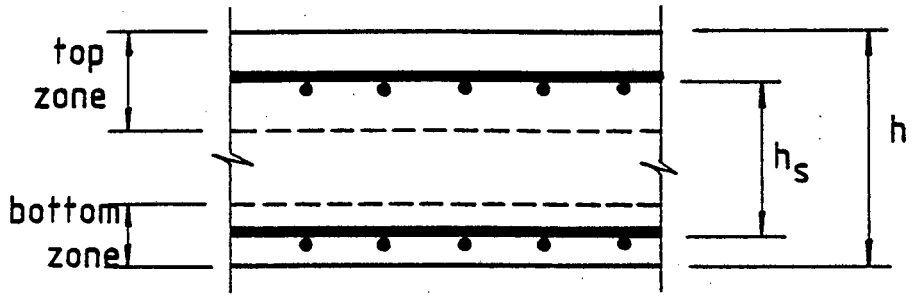


Figure 8.4 : Top and bottom zones for ultimate load calculations

Clark [48] suggests that the slab be divided into top and bottom regions with a thickness equal to twice the average cover to the top and bottom reinforcement respectively (Figure 8.4). The bending moment triad and in-plane forces are then transformed to equivalent in-plane forces in these two regions by :

$$N_{x\ top} = \frac{N_x}{2} + \frac{M_x}{h_s} \tag{8.11}$$

$$N_{y\ top} = \frac{N_y}{2} + \frac{M_y}{h_s} \tag{8.12}$$

$$N_{x\ bot} = \frac{N_x}{2} - \frac{M_x}{h_s} \tag{8.13}$$

$$N_{y\ bot} = \frac{N_y}{2} - \frac{M_y}{h_s} \tag{8.14}$$

$$N_{xy\ top} = \frac{N_{xy}}{2} + \frac{M_{xy}}{h_s} \tag{8.15}$$

$$N_{xy\ bot} = \frac{N_{xy}}{2} - \frac{M_{xy}}{h_s} \tag{8.16}$$

The required amounts of top and bottom steel in the predetermined reinforcing directions (Figure 8.3) and the concrete stresses can then be calculated from the following transformation of axial forces :

$$\bar{N}_x = N_x - 2\,N_{xy}\cot\alpha + N_y\cot^2\alpha + \left| \frac{N_{xy} - N_y\cot\alpha}{\sin\alpha} \right| \tag{8.17}$$

$$\bar{N}_\alpha = N_y\operatorname{cosec}^2\alpha + \left| \frac{N_{xy} - N_y\cot\alpha}{\sin\alpha} \right| \tag{8.18}$$

$$F_c = -2\,(N_{xy} - N_y\cot\alpha)\;(\cot\alpha \pm \operatorname{cosec}\alpha) \tag{8.19}$$

where the sign in the last bracket is the same as the sign of $(N_{xy} - N_y\cot\alpha)$

If $\bar{N}_x < 0$, then $\bar{N}_x = 0$, and

$$\bar{N}_\alpha = \operatorname{cosec}^2\alpha \left\{ N_y + \left| \frac{(N_{xy} - N_y\cot\alpha)^2}{(N_x - 2N_{xy}\cot\alpha + N_y\cot^2\alpha)} \right| \right\} \tag{8.20}$$

$$F_c = \frac{(N_x - N_{xy}\cot\alpha)^2 + (N_{xy} - N_y\cot\alpha)^2}{N_x - 2N_{xy}\cot\alpha + N_y\cot^2\alpha} \tag{8.21}$$

If $\bar{N}_\alpha < 0$, then $\bar{N}_\alpha = 0$, and

$$\bar{N}_x = N_x - 2\,N_{xy}\cot\alpha + N_y\cot^2\alpha + \left| \frac{(N_{xy} - N_y\cot\alpha)^2}{N_y} \right| \tag{8.22}$$

$$F_c = N_y + \frac{N_{xy}^2}{N_y} \tag{8.23}$$

If $\bar{N}_x < 0$, and $\bar{N}_\alpha < 0$, then $\bar{N}_x = 0$, $\bar{N}_\alpha = 0$ and

$$F_c = \tfrac{1}{2}(N_x + N_y) - \tfrac{1}{2}\sqrt{(N_x - N_y)^2 + 4N_{xy}^2} \qquad (8.24)$$

8.6 Principal stress calculation

The principal stresses in the top and bottom fibres of the slab under service loads give a good indication of the performance of the materials that constitute the slab.

Principal tensile stresses indicate whether the concrete has cracked. This is useful in assessing the class of structure as envisaged in the design codes (CP 110 [49] or SABS 0100 [50]). Crack widths can also be calculated from these stresses.

The stress distribution through the thickness of the slab is indicated in Figure 8.5.

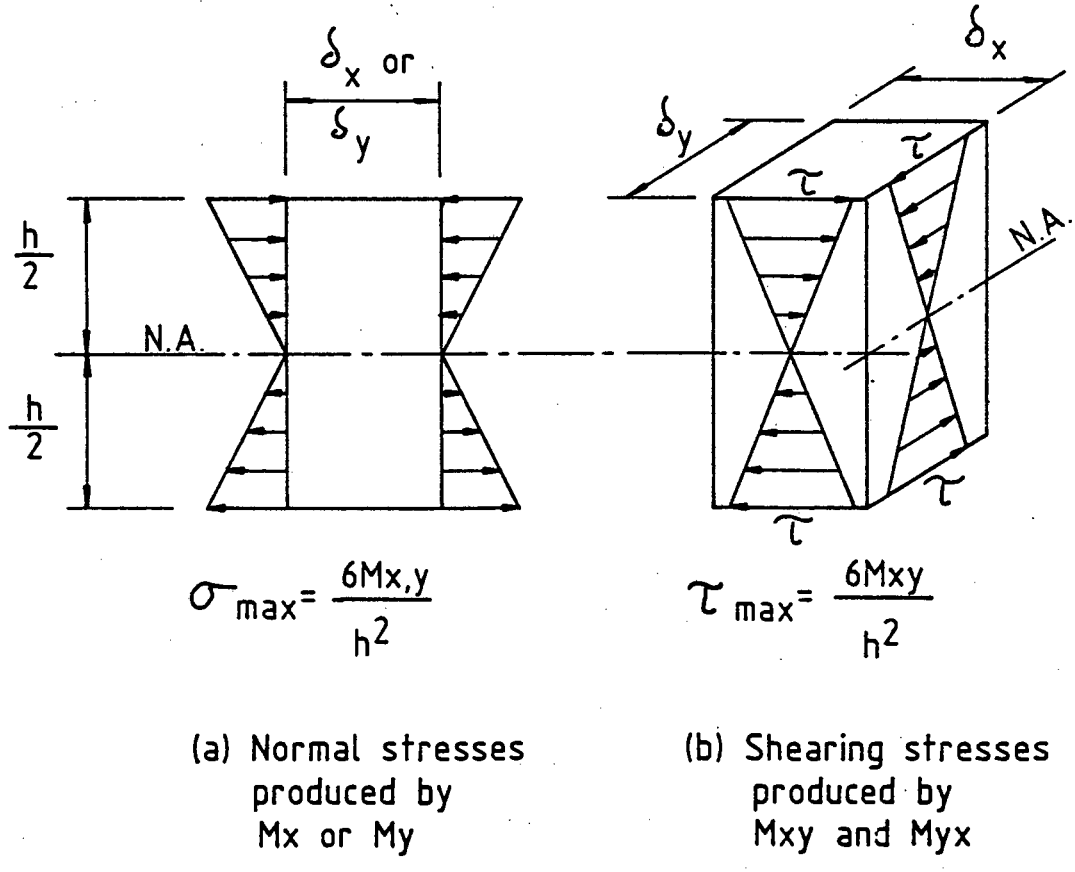


Figure 8.5 : Distribution of stresses through the slab

Top- and bottom fibre stresses in the x and y directions are then :

Top fibre :

$$\sigma_x = \frac{N_x}{h} - \frac{6\,M_x}{h^2} \tag{8.25}$$

$$\sigma_y = \frac{N_y}{h} - \frac{6\,M_y}{h^2} \tag{8.26}$$

$$\tau_{xy} = \frac{N_{xy}}{h} + \frac{6\,M_{xy}}{h^2} \tag{8.27}$$

Bottom fibre :

$$\sigma_x = \frac{N_x}{h} + \frac{6\,M_x}{h^2} \tag{8.28}$$

$$\sigma_y = \frac{N_y}{h} + \frac{6\,M_y}{h^2} \tag{8.29}$$

$$\tau_{xy} = \frac{N_{xy}}{h} - \frac{6\,M_{xy}}{h^2} \tag{8.30}$$

Principal stresses in the top or bottom fibres can be calculated from :

$$\sigma_{I,II} = \tfrac{1}{2}(\sigma_x + \sigma_y) \pm \tfrac{1}{2}\sqrt{(\sigma_x - \sigma_y)^2 + 4\,\tau_{xy}^2} \tag{8.31}$$

and the direction of $\sigma_{I,II}$ relative to the x-axis from :

$$\tan 2\theta = \frac{2\,\tau_{xy}}{\sigma_x - \sigma_y} \tag{8.32}$$

CHAPTER 9

VERIFICATION EXAMPLES

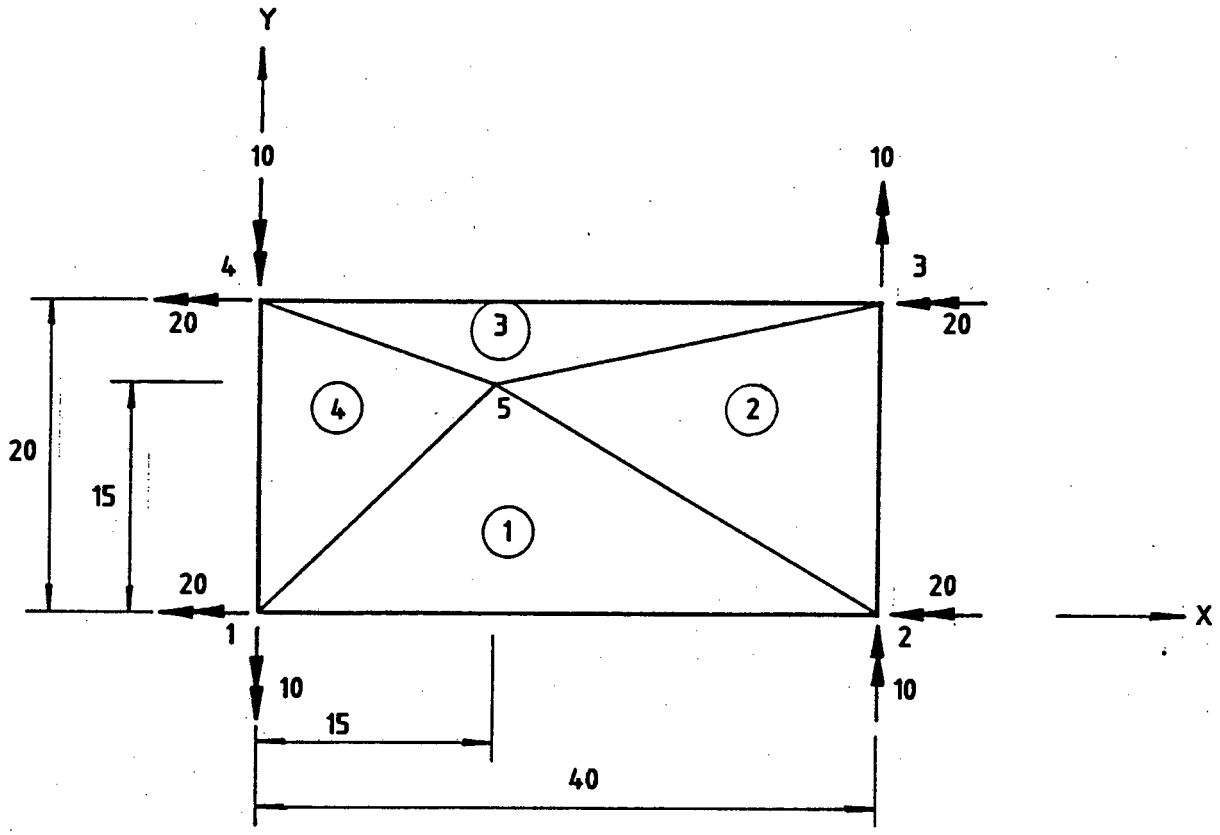
9.1 Introduction

Various examples are presented in this chapter to verify the validity of the modules that make up "PRESLAB". The plate bending examples are as suggested by Robinson [51] and presented by Batoz [7].

9.2 Plate bending examples

9.2.1 Patch test

The patch test considers a rectangular plate with boundary conditions and loads that result in a constant state of bending moments over the whole plate. The problem is illustrated in Figure 9.1.



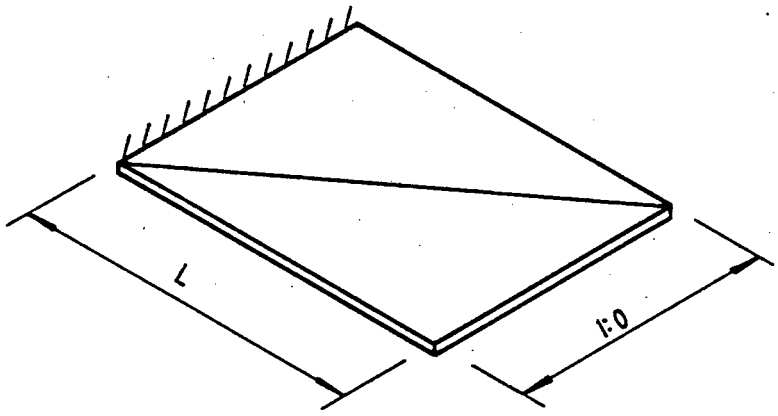
BOUNDARY CONDITIONS
u = 0 at nodes 2,3
v = 0 at nodes 1,2
w = 0 at nodes 1,2,4

Figure 9.1 : Patch test problem

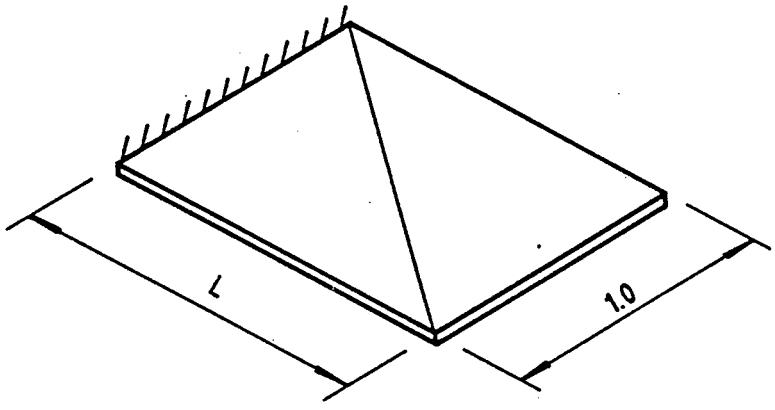
Node 5 can be located at any arbitrary position within the rectangle. For the material properties and geometry in Figure 9.1, "PRESLAB" returns the correct values for the bending moments at the centroid of each element, i.e. Mx = My = Mxy = 1.0.

9.2.2 Centilever rectangular plate under twisting load

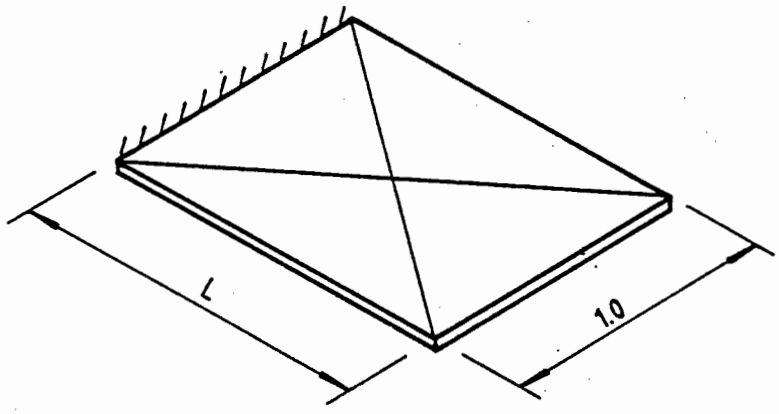
A very stringent test of the behaviour of triangular elements is the case of a rectangular plate under twist moments with one side fully clamped. Differential bending is obtained by applying differential loads or twisting moments at the two cantilever points on the plate. Three mesh orientations were investigated as indicated in Figure 9.2. The deflection w is plotted for increasing values of the element aspect ratio.



MESH A



MESH B



MESH C

Figure 9.2 : Rectangular cantilever plates

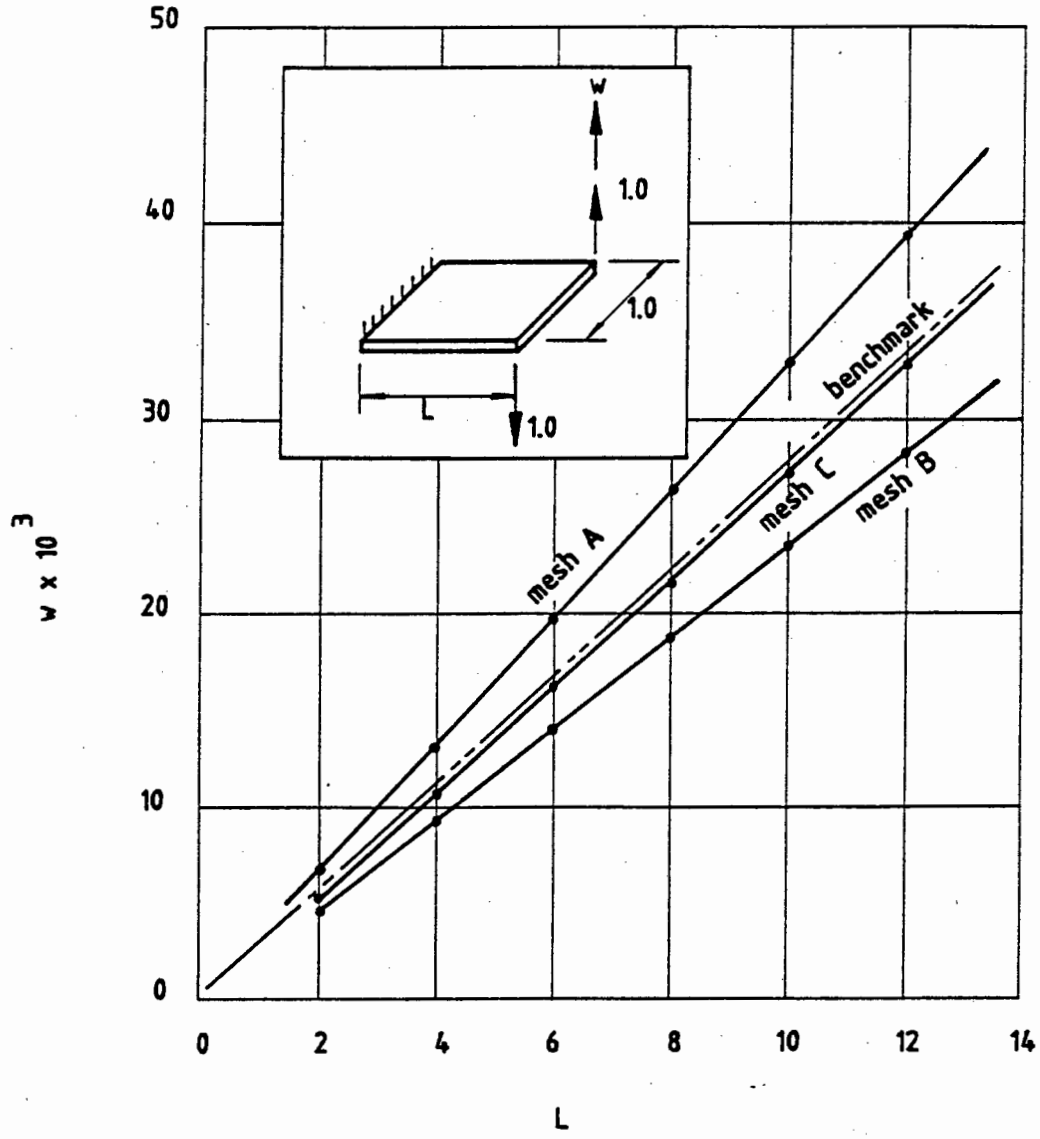


Figure 9.3 : Tip deflection of rectangular cantilever plate under differential loads (w vs L)

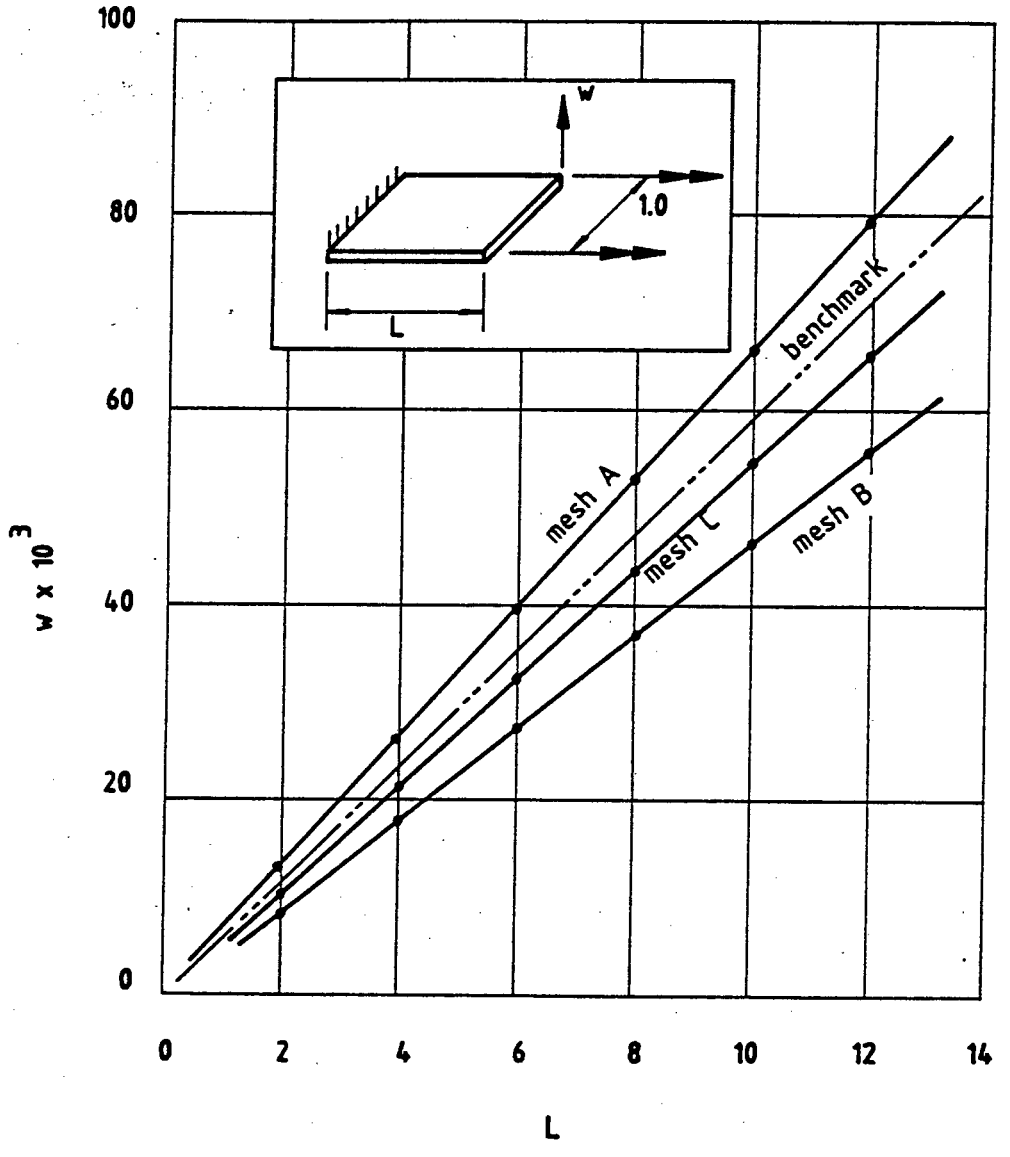


Figure 9.4 : Tip deflection of rectangular cantilever plate under twisting moments (w vs L)

The DKT element performs extremely well even at high aspect ratios.

Batoz [7] presents further benchmark tests that verify the validity of the DKT formulation. For the purposes of this thesis, the two tests listed above were deemed to be sufficient to validate the code for the plate bending element.

9.3 Plane stress examples

The formulation and performance of the constant strain triangle is extensively documented. Two tests were carried out to confirm the formulation of the CST element in "PRESLAB".

9.3.1 Direct stresses

Consider the rectangular plate in Figure 9.5. Under the boundary conditions and loading conditions indicated, "PRESLAB" returns the correct values for the in-plane stresses, i.e. : Nx = Ny = -1.0 and Nxy = 0.0.

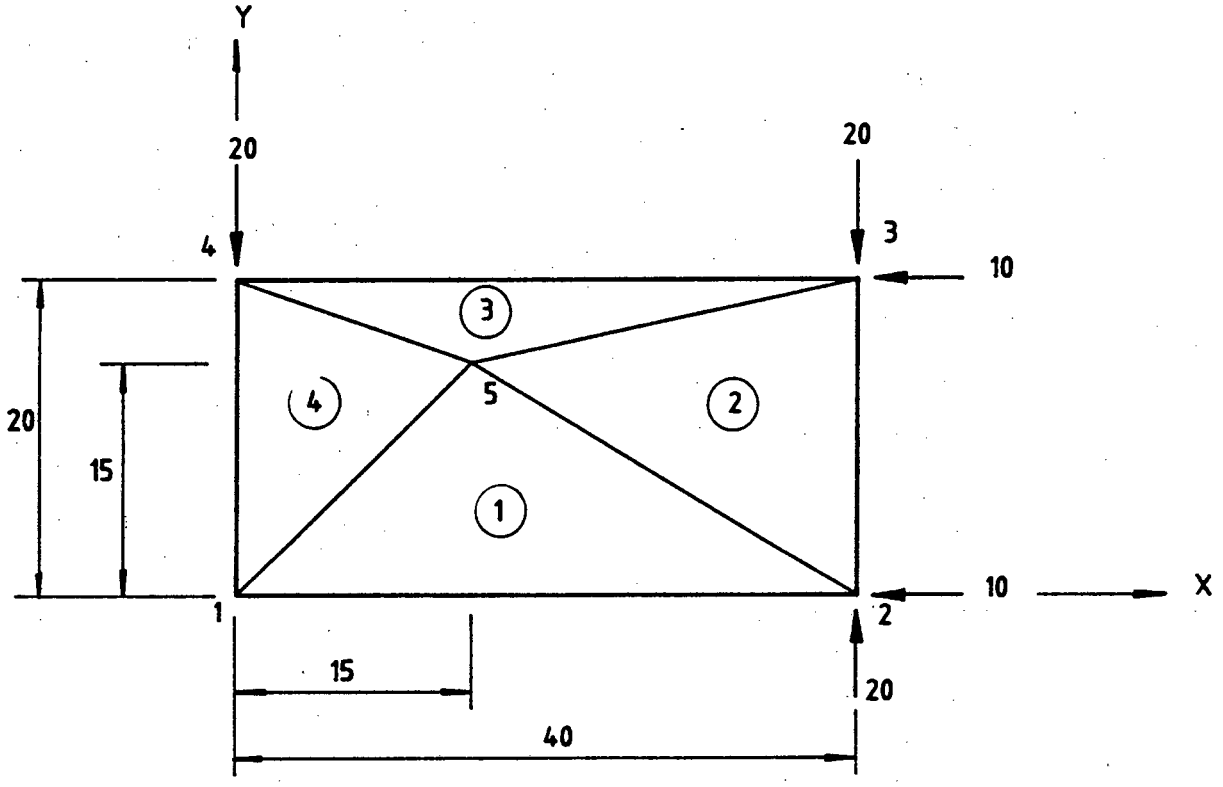
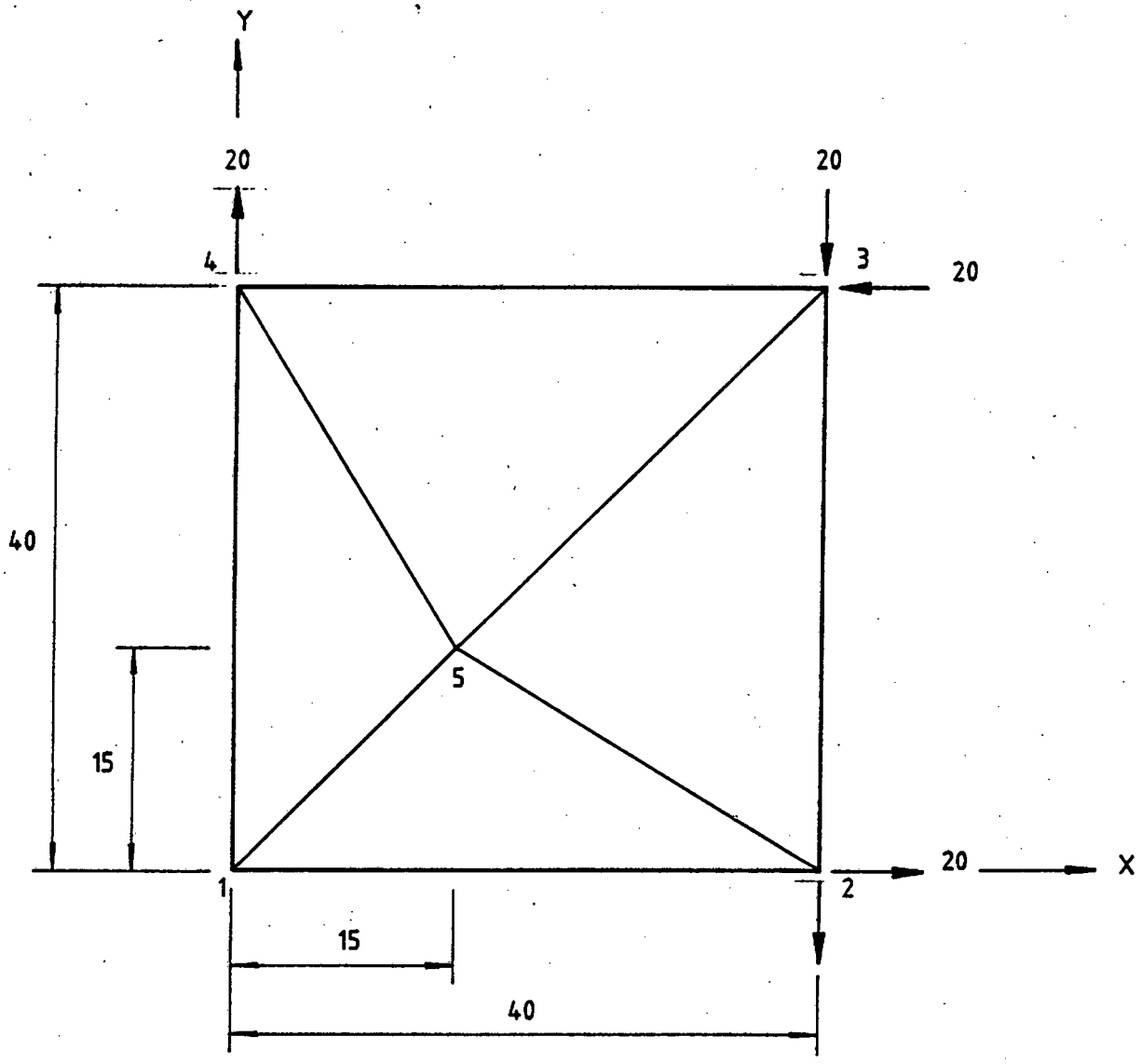


Figure 9.5 : Rectangular plate under constant plane stress

9.3.1 Shear stresses

Consider the rectangular plate in Figure 9.6. Under the boundary conditions and loading conditions indicated, "PRESLAB" returns the correct values for the in-plane stresses, i.e. : Nx = Ny = 0.0 and Nxy = -1.0.



BOUNDARY CONDITIONS

u = 0 at nodes 1,4
v = 0 at node 1
w = 0 at nodes 1,2,3,4

Figure 9.6 : Rectangular plate under constant shear stress

10.2 Defining the slab

Elements and nodes in the finite element model of the slab are generated by defining four super-elements (Figure 10.2) through EDSUP and triangulated through SUPCRO.

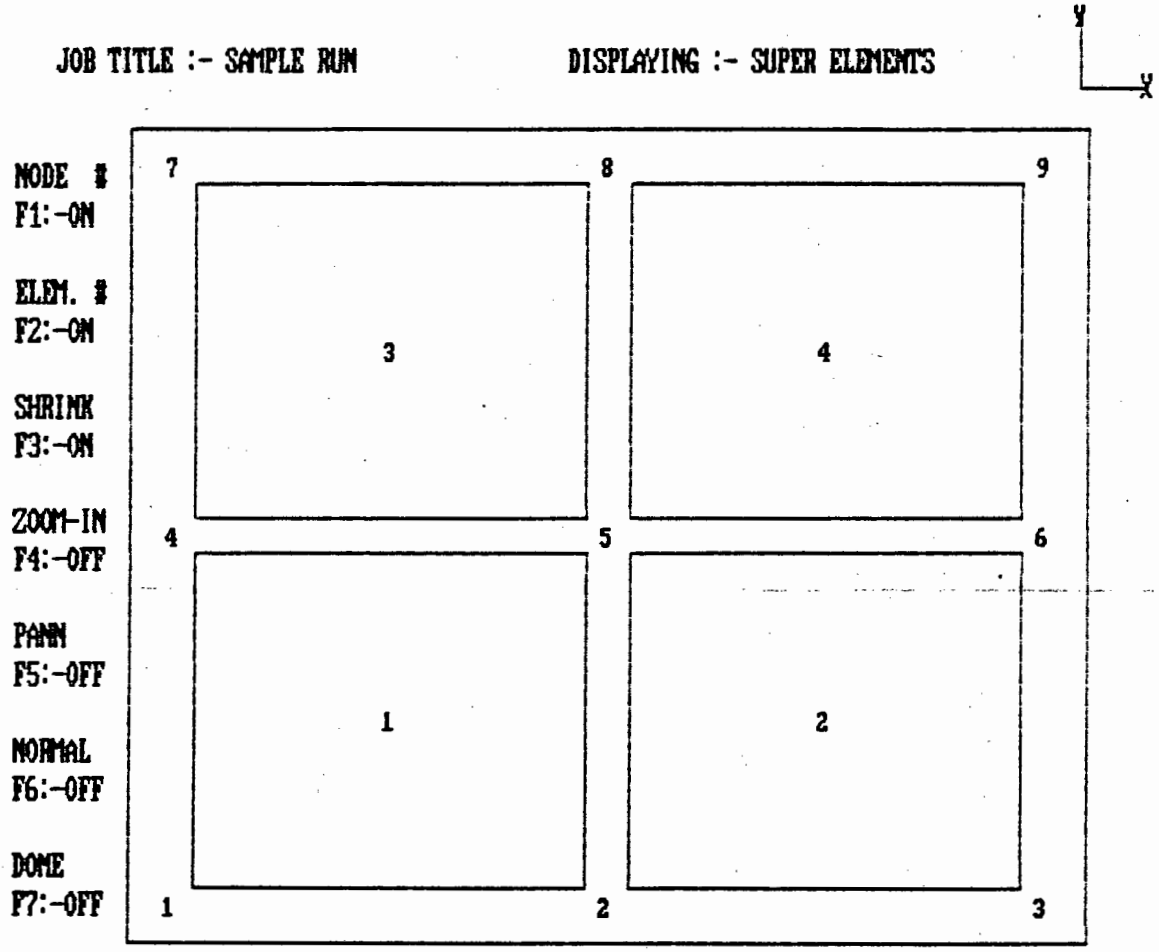


Figure 10.2 : Super-element model of the slab

The generated finite element model is displayed in Figure 10.3.

Fixed nodes and material properties are supplied through the EDSYS editor.

10.3 Setting up and triangulating the structure stiffness matrix

After defining the model, the data is checked (ACTIVE) and the structure stiffness matrix calculated (MATRIX). This stiffness matrix is then reduced to the upper triangular form (TRIANG).

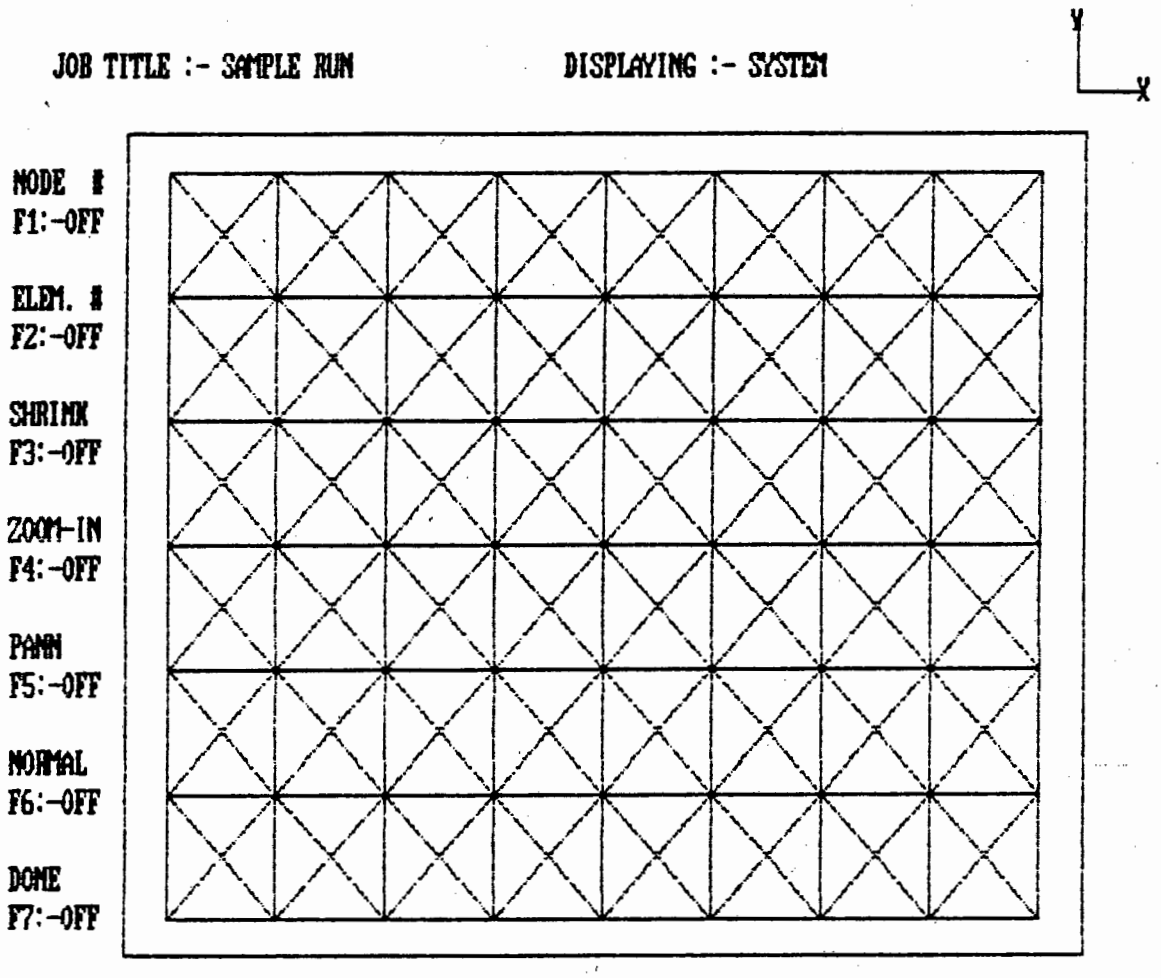


Figure 10.3 : Finite element model of the slab

10.4 Load cases

10.4.1 Basic loads

The following six basic load cases are defined (LOADED), the load vectors set up (LODVEC) and nodal displacements calculated (DEFLEC) :

i) Self weight - Self weight is automatically calculated from the weight density supplied and the element volumes calculated.

ii) Imposed load - Imposed loads of 3 kN/m² are applied over the entire area of the slab.

iii) Area loads - Unit area loads are applied over each quarter of the slab. These unit loads.will be used to calculate patterned loading at the ultimate limit state.

10.4.2 Prestressing loads

It is intended that prestressing be applied in both the longitudinal and transverse directions of the slab. From the profile module (PROFIL), the loading due to a single prestressing cable can be calculated, i.e. :

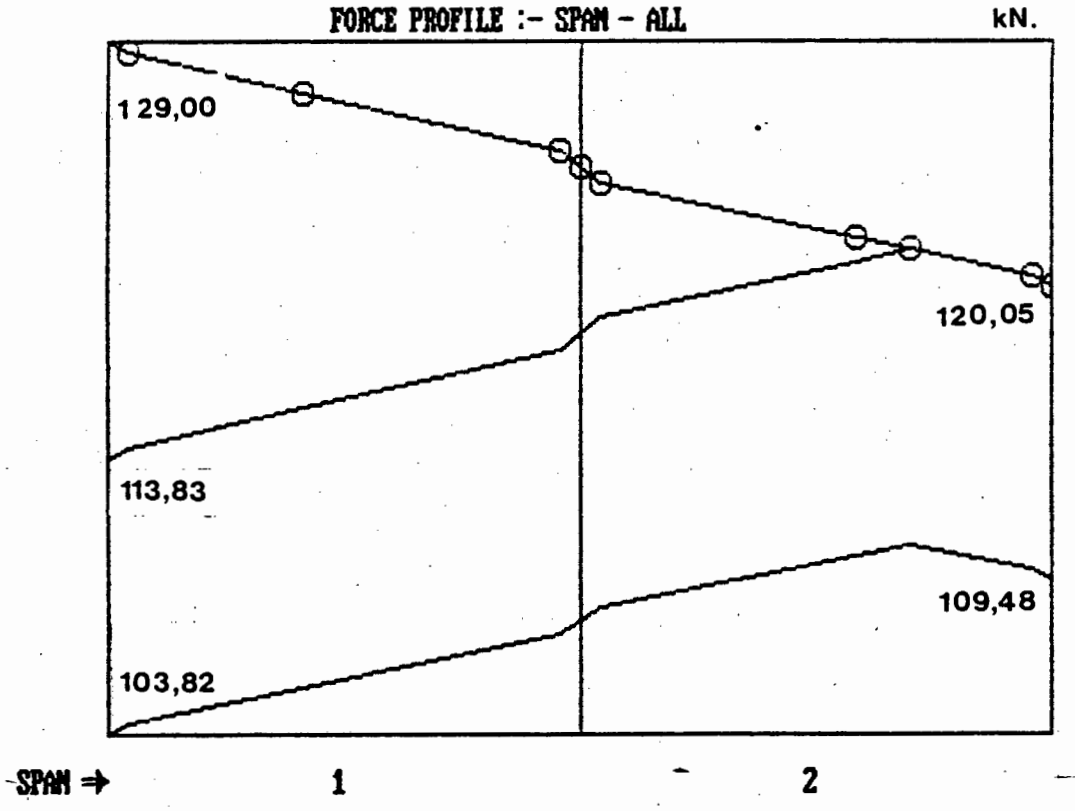


Figure 10.4 : Force profile of prestressing cable (longitudinal direction)

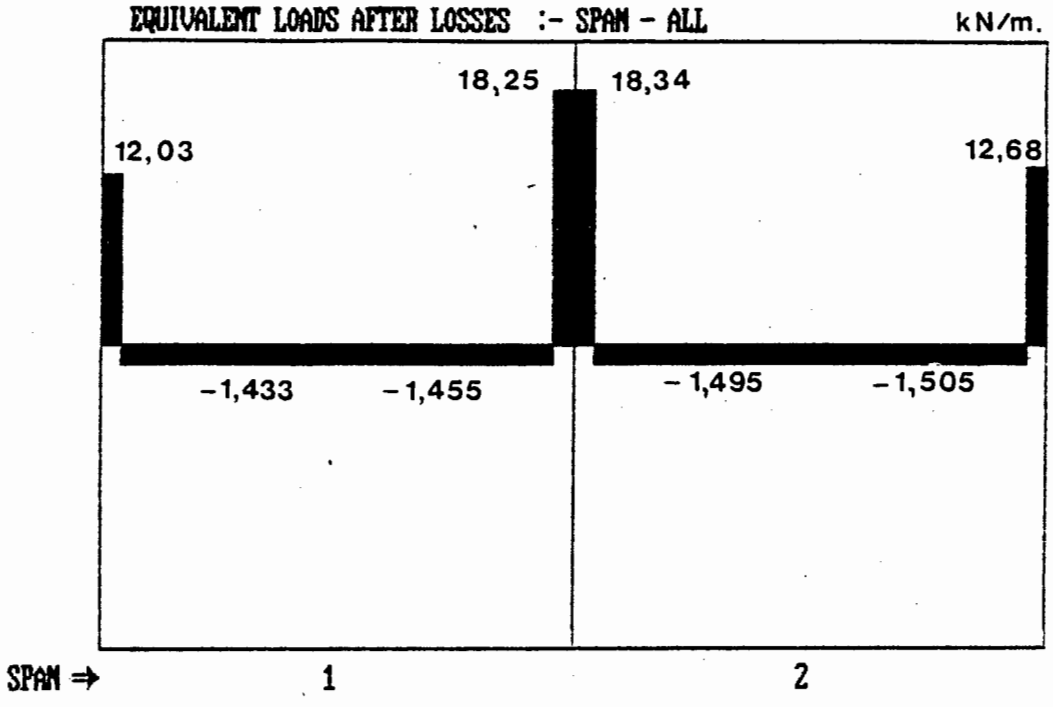


Figure 10.5 : Equivalent loads due to prestressing cable after long term losses (longitudinal direction)

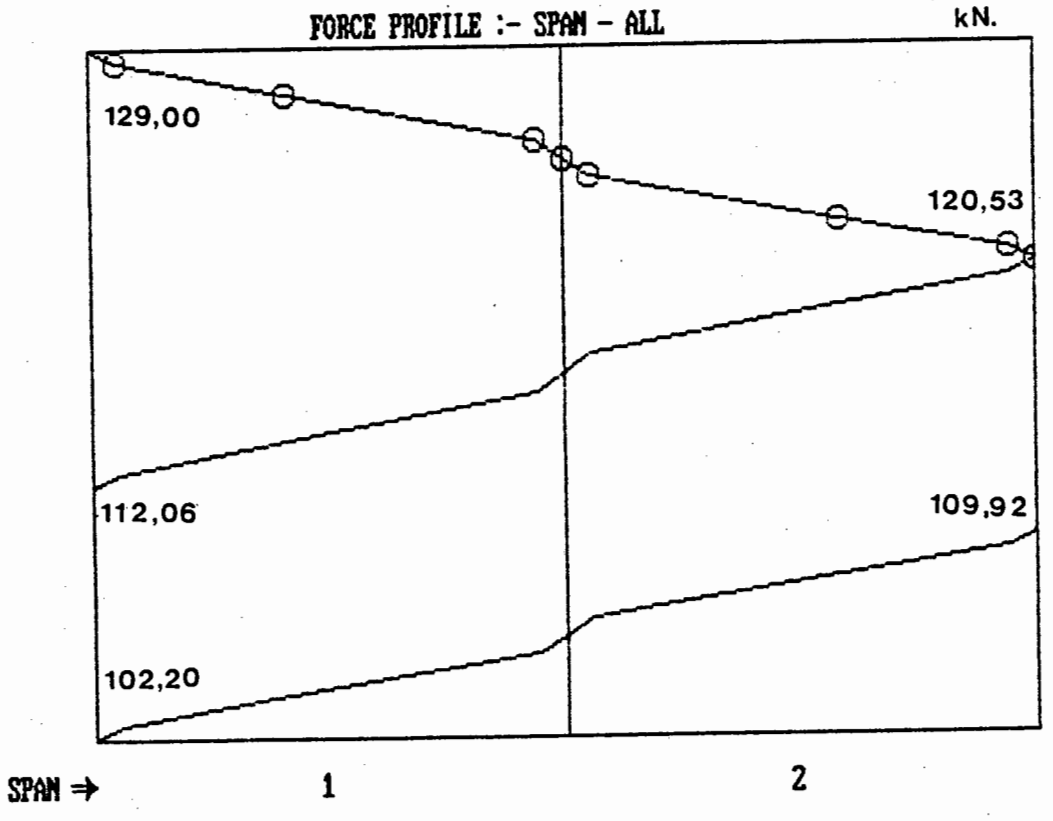


Figure 10.6 : Force profile of prestressing cable (transverse direction)

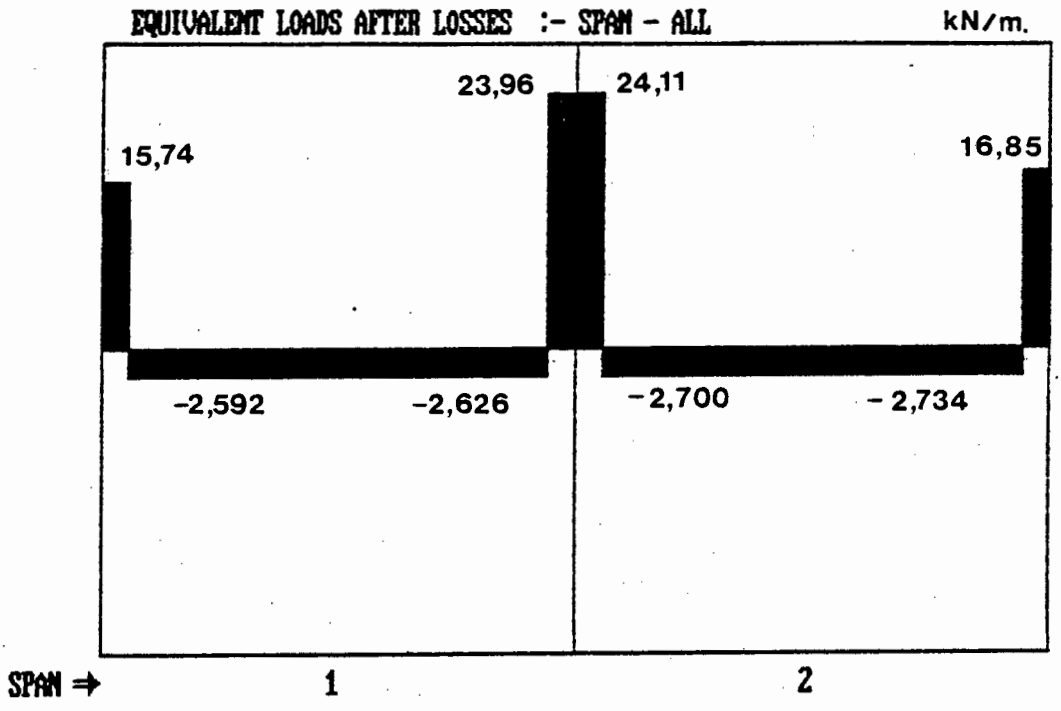


Figure 10.7 : Equivalent loads due to prestressing cable after long term losses (transverse direction)

Initially, it is assumed that :

- The prestressing cables will be spaced at 1.0m c/c in both the longitudinal and transverse directions.

- The prestressing cables will be stressed alternately from each end.

The following loads will then result from the prestressing cables :

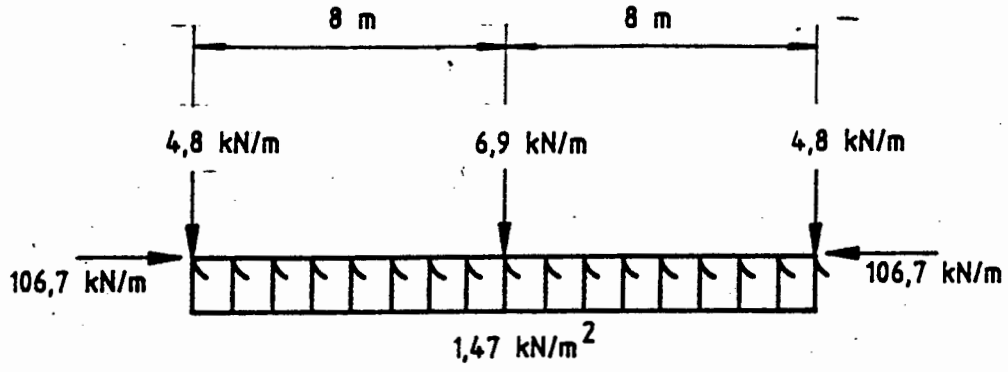


Figure 10.8 : Longitudinal direction
loads due to prestressing cables at 1.0m c/c

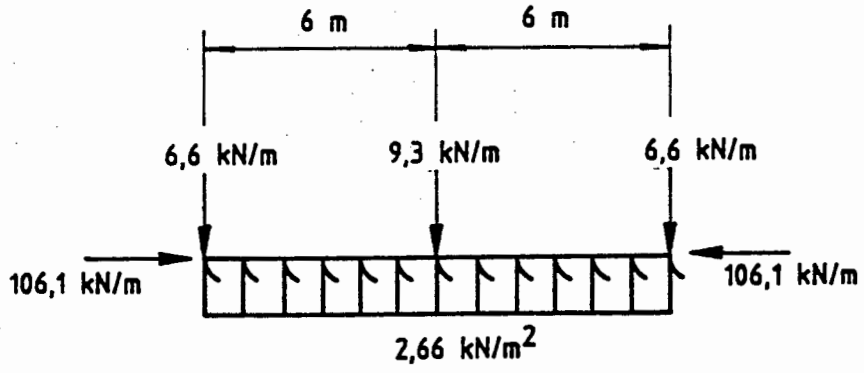


Figure 10.9 : Transverse direction
loads due to prestressing cables at 1.0m c/c

The loads due to prestressing can now be defined (LOADED), the load vectors set up (LODVEC) and the nodal displacements calculated (DEFLEC).

10.5 Load balancing

As pointed out in Chapter 8, the term "load balancing" is actually a misnomer. "Deflection balancing" would be a more apt description. Our aim is then to apply equivalent transverse loads due to prestressing on the slab to lift out the deflections due to applied long term loads. This process can be approached visually as described in the following paragraphs.

Assume that 10% of the imposed load will be applied to the structure on a long term basis. Self weight and imposed loads are combined (COMBI) to form the new load case, i.e. :

Long term loads = 1.0 x self weight + 0.1 x imposed load.

The deflected shape of the slab under this load case appears as follows :

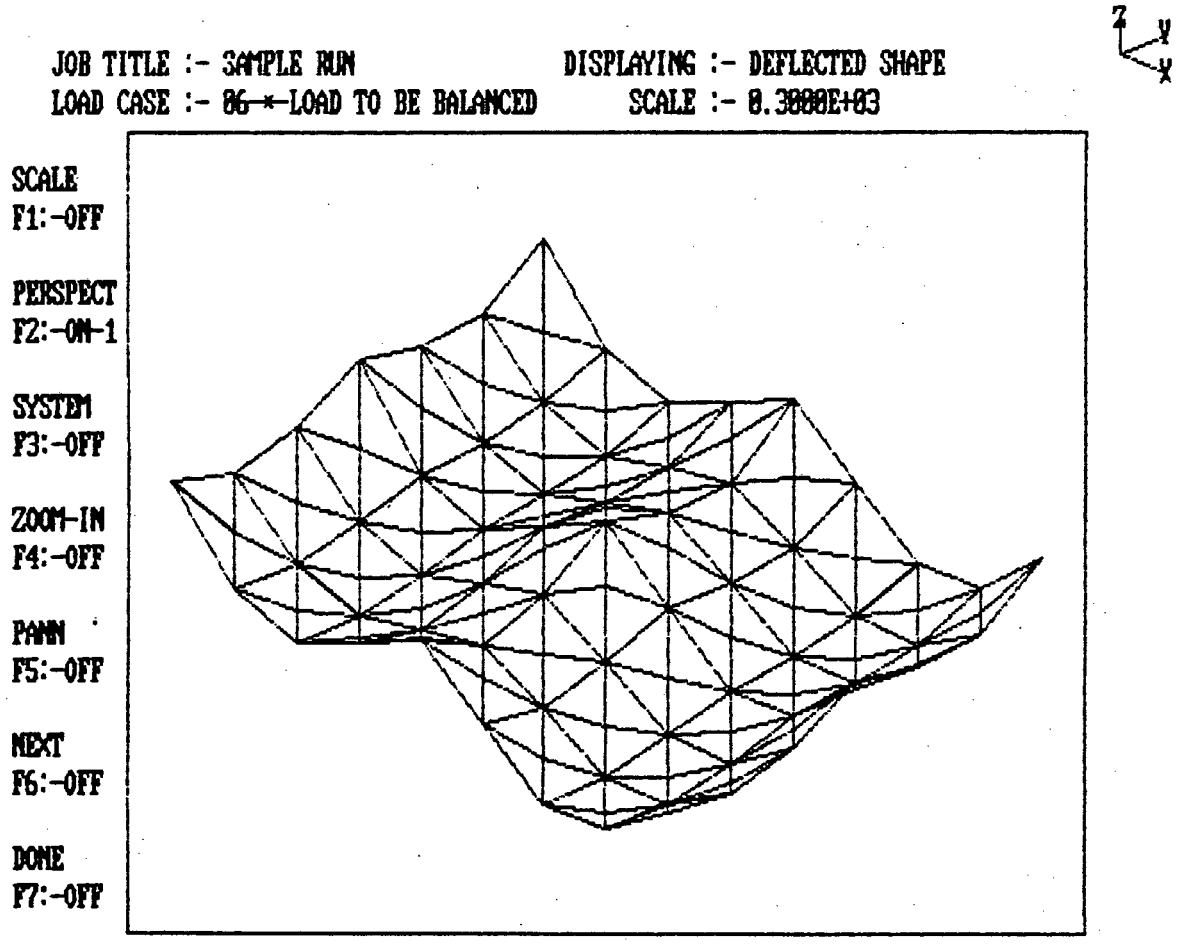


Figure 10.10 : Deflected shape of the slab under long term loads

Long term loads can now be combined with multiples of the prestressing load cases with the aim of reducing the deflections in the plate. For the longitudinal cables spaced at 350mm c/c and the transverse cables spaced at 1 000mm c/c, the load combinations will be :

Balanced loads = 1.0 x long term loads +
2.85 x longitudinal equivalent loads +
1.0 x transverse equivalent loads.

Under this load combination, the deflected shape of the slab appears as follows :

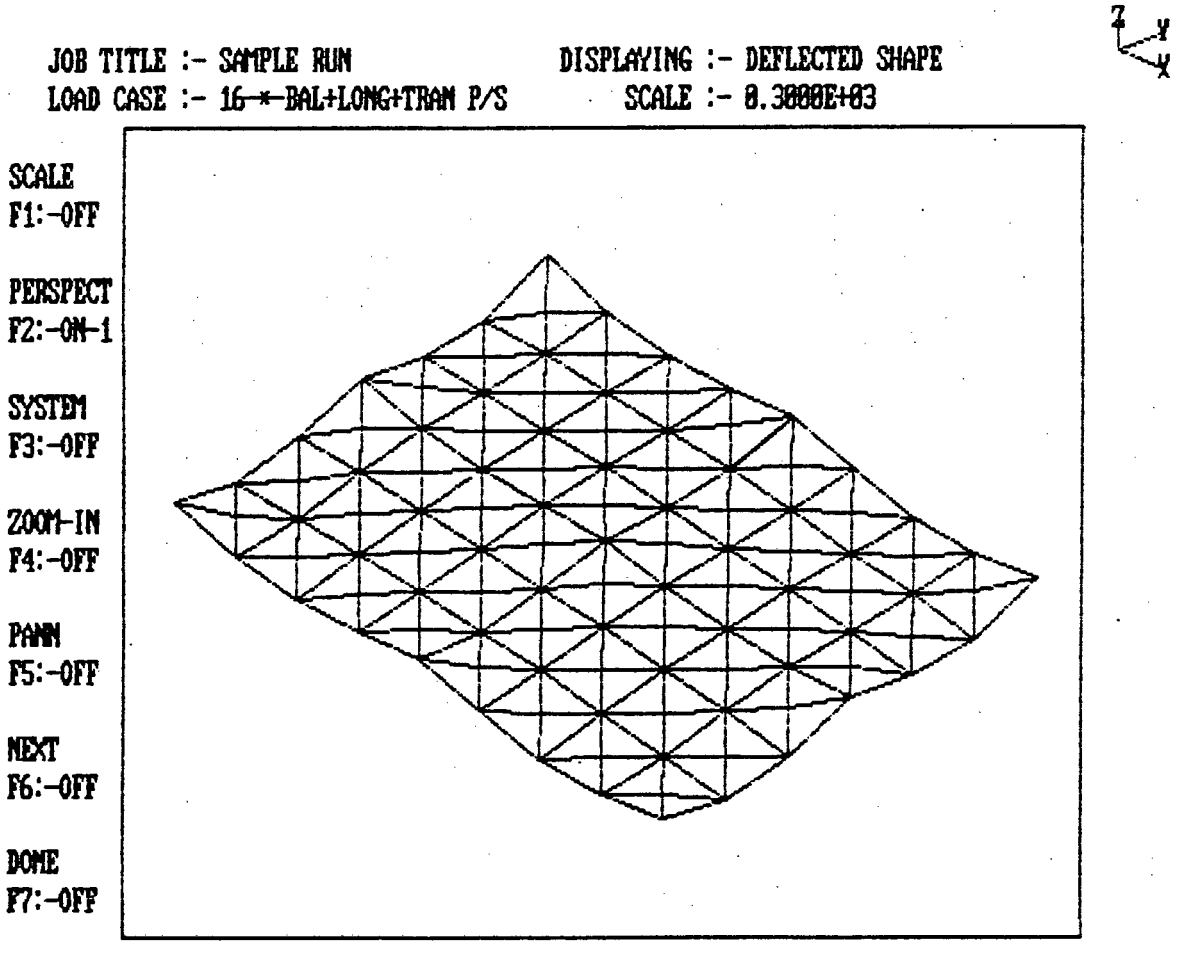


Figure 10.11 : Deflected shape of the slab under long term loads + prestressing

It is clear that most of the deflections have been balanced by the prestressing. Additional prestressing cables will, however, be required along the column strips of the slab to lift out some local deflections. Load cases can now be set up and deflections calculated for a single cable along each of the six column strips in Figure 10.1 (1,2,3,A,B and C).

Combining multiples of the column strip loads with the "long term + prestressing" load case, we find that, with the additional cables listed below, the slab has virtually no deflection (Figure 10.12).

- Strip 1 : - 4 additional cables
- Strip 2 : - 2 additional cables
- Strip 3 : - 4 additional cables
- Strip A : - 2 additional cables
- Strip B : - 1 additional cable
- Strip C : - 2 additional cables.

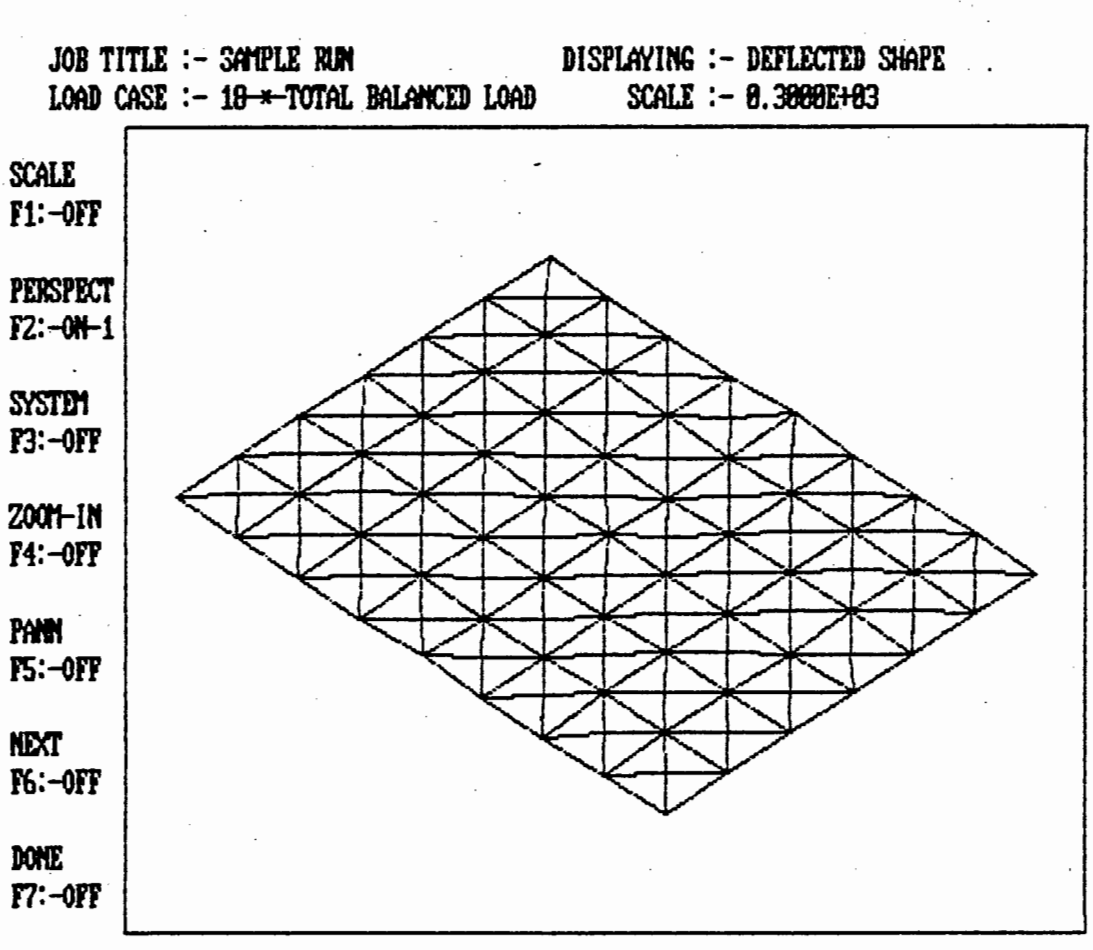
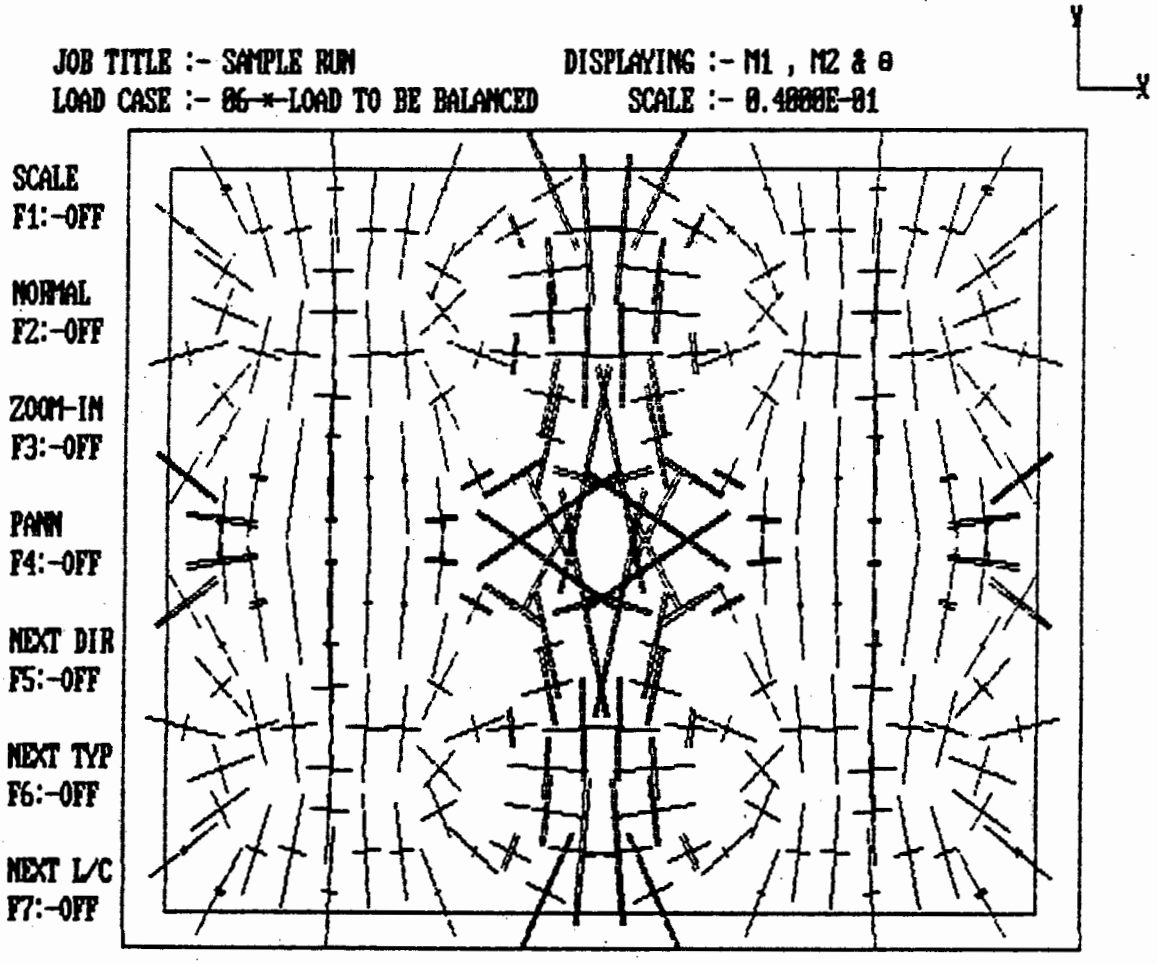


Figure 10.12 : Final deflected shape of the slab under long term loads + final prestressing

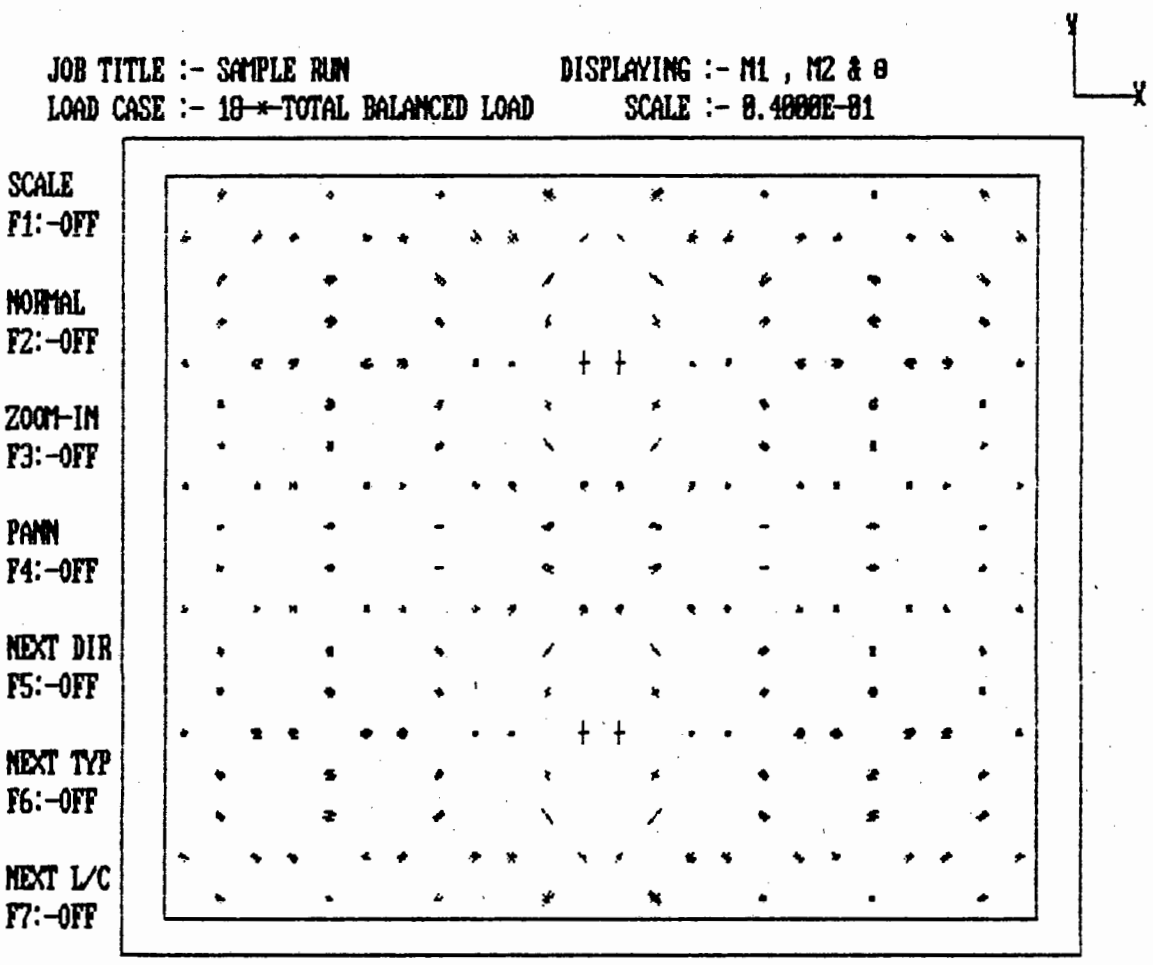
Viewing the vectors of the principal moments in the slab clearly shows that the slab is now virtually in a state of plane stress. The maximum moments under the two load cases differ by a factor of 6.



M_{max} = 46.42 kNm/m

M_{min} = -26.83 kNm/m

Figure 10.13 : Vector plot of principal moments under long term loads



M_{max} = 2.97 kNm/m

M_{min} = -4.51 kNm/m

Figure 10.14 : Vector plot of principal moments under long term loads + final prestressing

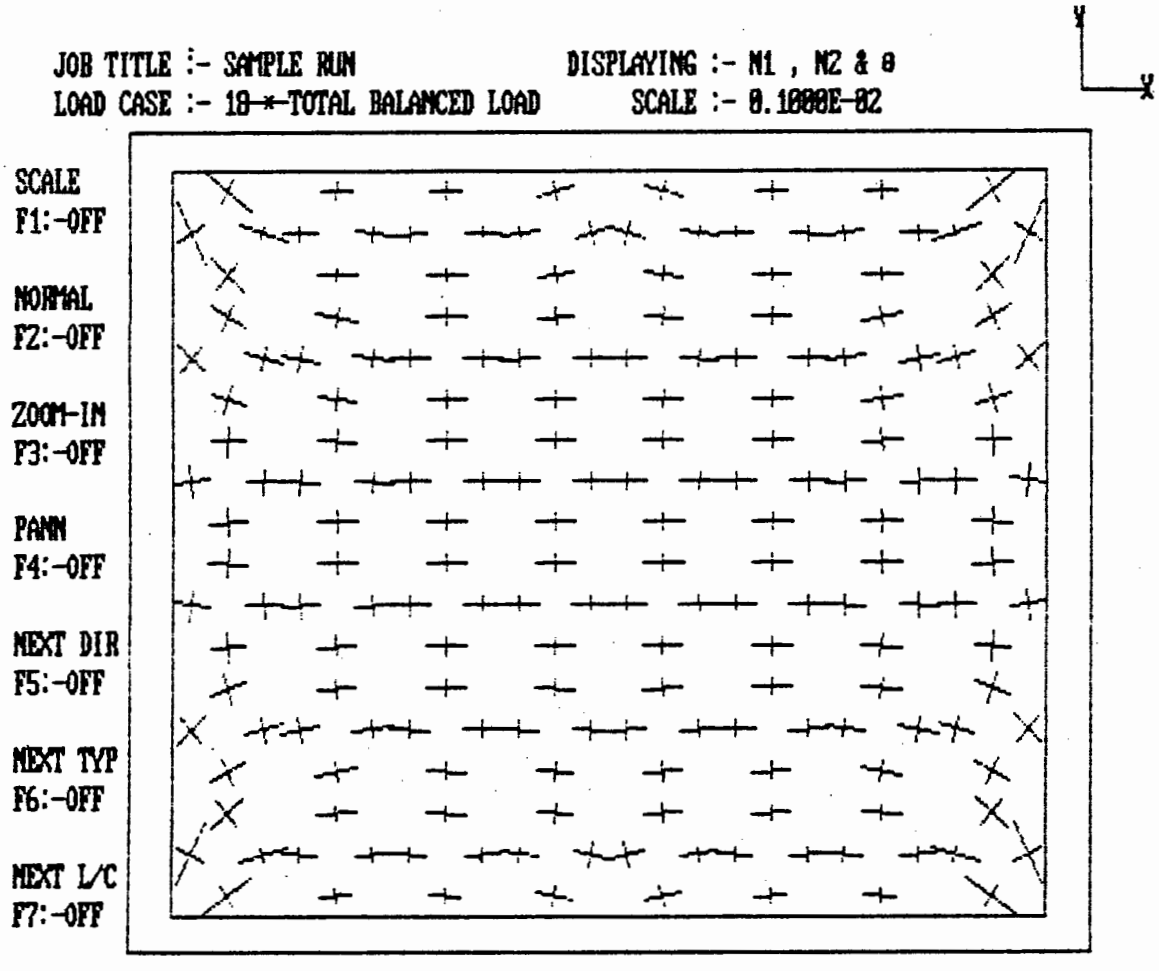


Figure 10.15 : Vector plot of principal in-plane forces under long term loads + final prestressing

10.6 Ultimate limit state - design of untensioned reinforcement

Having determined the prestressing required, the various load cases can now be combined with load factors to determine the design stresses. Patterned loading at the ultimate limit state can also be applied by factored combination of the various unit area loads. Untensioned steel can then be designed in any direction using the DESIGN module.

Wood/Armer [40,41] moments can be used to design untensioned steel by the load factor method. These moments can also be used in the limit state design of untensioned steel for slabs with no in-plane forces. Alternatively, untensioned steel requirements at the ultimate load state can be calculated from Clark's [48] recommendations.

Consider only the following load combination at the ultimate limit state :

- Ultimate load = 1.4 x Self weight +
  1.0 x Prestressing (long term) +
  1.6 x Imposed load (over the whole slab).

From the design module (DESIGN) the maximum untensioned steel area required and concrete stress is :

- Steel area = 1 195 mm^2/m
- Concrete stress = 14.8 MPa.

For the same loads at the serviceability limit state, the maximum and minimum principal stresses in the extreme fibres of the plate are :

- Maximum principal stress = 2.9 MPa (compressive)
- Minimum principal stress = 0.2 MPa (compressive).

It is clear that the slab remains uncracked under this load case.

Punching shear stresses can now be calculated from the reactions and the design completed.

CHAPTER 11

CONCLUSIONS

11.1 Concluding remarks

Prestressed concrete flat slabs are usually analysed and designed by the equivalent frame method [45-47]. Slabs are divided into column and middle strips. These strips are then designed as prestressed beams utilising the distribution factors for ordinary reinforced concrete flat slabs proposed in the design codes [49,50]. Using this approach, the effects of stiff walls, penetration and odd shaped slabs are extremely difficult to predict. "PRESLAB" offers an alternative analysis and design procedure which looks at the behaviour of the entire slab under loading and prestressing.

The data editors (EDSUP, EDSYS and LOADED) and super-element section, perform a vital task in rapid data generation. Data errors are usually made when this task is performed manually and account for the major proportion of abortive work in finite element analyses. Using the super-element sections, data editors and plotting routines (PLOTSUP and PLOTSYS) minimizes the risk of making obvious and gross data errors.

"PRESLAB" models the bending and in-plane behaviour of a plate by superimposing the traingular discreet Kirchhoff plate bending element and the constant strain traingle plane stress element in the same formulation. This formulation allows the designer to analyse both in-plane and bending behaviour of prestressed flat slabs on the same element. This concept is certainly not new, but to the author's knowledge, very few programs have been developed along these lines. With prestressed flat slabs gaining ever increasing popularity and acceptance among designers and clients, it is to be expected that the demand for this type of analysis capability will increase.

The discreet Kirchhoff (DKT) element is one of the most efficient triangular plate bending elements available today [6]. The explicit formulation in global coordinates used in "PRESLAB" is not the fastest of three different formulations tested but does

require, by far, the least number of lines to code. Limited internal memory on micro-computers makes this formulation the most suitable. As micro-computers with more in-core memory become freely available and less costly, the other formulations may become more attractive.

The constant strain traingle (CST) element used to model the in-plane effects of prestressing is renowned for its relatively poor representation of the plane stress condition. In-plane loading due to prestressing is usually quite well distributed throughout prestressed flat plates. From a design point of view, the CST element is accepted as the most suitable plane stress element to be used in conjunction with the DKT element where bending behaviour dominates the solution. The CST element is extremely quick and easy to program. Considerations of speed and limited in-core memory on micro-computers made the CST element an obvious choice for this application.

An out-of-core solver [27] was developed for the solution of the sets of linear equations resulting from the finite element analysis. This solver was designed specifically for micro-computers and is based on the well known equation solver developed by Mondkar and Powell [24]. The solver operates on a coefficient matrix which is stored out-of-core as a one dimensional array of values under the skyline.

The coefficient matrix is decomposed into an upper triangular matrix by the Cholesky method. Improvements on the Mondkar/ Powell [24] solver pertain to the partitioning of the coefficient matrix into blocks of reduced and unreduced terms. The size of these blocks vary as the decomposition of the matrix proceeds. This dynamic form of partitioning selects an optimum size of the blocks and leads to substantial savings in terms of execution time and data transfer from auxiliary storage.

Because prestressed concrete slabs can now be analysed as a unit, the design approach must necessarily change. The "load balancing" technique as proposed by Lin [36] offers an efficient and quick method of assessing the effects of prestressing on slabs. With this technique, the self weight, dead load and a proportion of the imposed load is balanced by "equivalent" transverse loads due to prestressing to yield a condition of zero deflection in the slab. Under this serviceability limit state, the slab is in a state of plane stress. Untensioned steel need only be designed to take account of the loads that differ from the "balanced" load.

"Equivalent" loads due to prestressing are calculated by the PROFIL module. This module also calculates the prestressing cable geometry and the prestressing forces along the length of the cable.

The deflected shape plotting module (PLOTDEF) allows a quick visual assessment of the effects of loading and prestressing. A condition of zero deflection in the plate can now be achieved with the minimum of calculation and a lot more confidence. This visual representation of the deflections in the slab also contributes to the designers understanding of the behaviour of the slab under various loading conditions and must lead to a more rational design.

Bridge engineers have long researched and proposed procedures for the design of untensioned steel in flat slabs subjected to in-plane forces and a triad of moments [43,48].

To aid the design the untensioned steel required at the serviceability or ultimate limit states, Wood/Armer [40,41] moments and untensioned steel design as recommended by Clark [48] was incorporated in the program. Principal stresses in the top and bottom fibres of the plate can also be calculated.

The analysis of bridge decks is normally carried out using influence surfaces and design tables for a number of very specific plate configurations and boundary conditions. "PRESLAB" now offers the capability of accurately analysing the most complex slab geometry very quickly.

The vector plot module, PLOTVAL, plots vectors of the moments and in-plane forces. These plots contribute to the designers understanding of the behaviour of the slab and allows a quick assessment of the effects of various loading conditions.

11.2 Scope for further research

The constant strain triangle (CST) used in "PRESLAB" to model the in-plane effects of prestressing seems to detract from the efficiency of the entire solution. Because the in-plane strains are constant over the element, bending and in-plane stresses are only calculated at the element centroid. Research [6,7] has shown that the DKT bending element is capable of producing reliable results at a number of points on the element. Combining the DKT element with a formulation of the plane stress element which produces reliable results at the same points on the element would mean that fewer elements could produce the same number of results.

A formulation of the DKT element with direct superposition of the linear strain triangle (LST) [13] and the development of newer quadrilateral shell elements [14,52] is very promising. The use of these elements for micro-computer analysis of prestressed flat slabs certainly warrants further research.

The DKT element has proved to be extremely robust and efficient. The application of this element in general shell analysis, non-linear problems and dynamic analysis on micro-computers also warrants further research.

11.3 Scope for further development

Various developments can be considered to improve and enhance the performance of "PRESLAB". These are listed below :

- Prescribed deflections at nodal points can be included to model support settlements.

- Elastic constraints at supports. Column stiffnesses can then be included in the analysis of the slab.

- Beam elements can be included in the slab model.

- Punching shear stress calculation can be performed directly from the reaction resultants.

- Creep, shrinkage and temperature effects can be included as load cases.

- Orthotropic material properties to model voided and ribbed slabs.

- A direct interface between the cable profile program and the load case section to apply the prestressing loads directly.

REFERENCES

[1] BATHE, K.J., "Finite element procedures in engineering analysis", Prentice Hall (1982).

[2] COOK, R.D., "Concepts and applications of finite element analysis" John Wiley & Sons (1981)

[3] HINTON, E. and OWEN, D.R.J., "Finite element programming", Academic Press (1977)

[4] BREBBIA, C.A. and CONNOR, J.J., "Fundamentals of finite element techniques", Butterworth & Co. (1973)

[5] NORRIS, C.H., WILBUR, J.B. and UTKU, S., "Elementary structural analysis", McGraw-Hill Kogakusha (1976)

[6] BATOZ, J.-L., BATHE, K.-J. and HO, L.-W., "A study of three-node triangular plate bending elements", Int. J. num. Meth. Engng., 15, 1771-1812 (1980)

[7] BATOZ, J.-L., "An explicit formulation for an efficient triangular plate-bending element", Int. J. num. Meth. Engng., 18, 1077-1089 (1982)

[8] JOSEPH, K.T. and RAO, K.S., "A fast algorithm for triangular plate bending element", Int. J. num. Meth. Engng., 14, 1100-1104 (1979)

[9] JEYACHANDRABOSE, C. and KIRKHOPE, J., "An alternative explicit formulation for the DKT plate-bending element", Int. J. num. Meth. Engng., 21, 1289-1293 (1985)

[10] OLSON, M.D. and BEARDEN, T.W., "A simple flat triangular shell element revisited", Int. J. num. Meth. Engng., 14, 51-68 (1979)

[11] CARPENTER, N., STOLARSKI, H. and BELYTSCHKO, T., "A flat triangular shell element with improved membrane interpolation", Department of Civil Engineering, Northwestern University, Evaston, Illinois, U.S.A.

[12] TALBOT, M. and DHATT, G., "Three discrete Kirchhoff elements for shell analysis with large geometrical non-linearities and bifurcations", Eng. Comput., 4 March 1987, pp. 15-22

[13] DHATT, G., MARCOTTE, L. and MATTE, Y., "A new triangular discrete Kirchhoff plate/shell element", Int. J. num. Meth. Engng., 23, 453-470 (1986)

[14] DHATT, G., MARCOTTE, L., MATTE, Y. and TALBOT, M., "Two new discrete Kirchhoff plate shell elements", 4th. Symp. Num. Meth. Engng., Atlanta, 599-604 (1986)

[15] NORTON, P., "Inside the IBM PC (access to advanced features and programming)", Robert J. Brady Co.

[16] NORTON, P., "(The Peter Norton) Programmers guide to the IBM PC", Microsoft Press, (1985)

[17] COFFRON, J.W., "Programming the 8086/8088", Sybex Inc., (1983)

[18] STARTZ, R., "8087 applications and programming for the IBM PC and other PCs", Robert J. Brady Co., (1983)

[19] GOSCHI, G. and SCHUELER, J.B., "WATFOR-77 Language Reference Manual", Watcom Publications Ltd., 1985

[20] "WATKOM GKS Graphics Tutorial and Reference Manual", Watcom Publications Ltd., 1986

[21] "IBM Personal Computer Professional FORTRAN", Ryan-McFarland Corp., 1984

[22] "Microsoft Macro Assembler for the MS-DOS Operating System", Microsoft Corporation, 1985

[23] PIRCHER, H., R.M. Suite of Programs, Technische Datenverarbeitung, Austria

[24] MONDKAR, D.P. and POWELL, G.H., "Large capacity equation solver for structural analysis", Computers & Structures, 4, 699-728 (1974)

[25] RECUERO, A. and GUTIEPREZ, J.P., "A direct linear system solver with small core requirements", Int. J. num. Meth. Engng., 4, 633-645 (1979)

[26] BETTES, P. and BETTES, J.A., "A profile matrix solver with built in constraint facility", Eng. Comput., 3, 209-216 (1986)

[27] DU TOIT, A.J. and DOYLE, W.S., "A dynamically partitioned, out-of-core skyline solver for micro-computers", Proc. 2nd Int. NUMETA Conf., University of Swansea, Wales (1987)

[28] REID, J.K. and HARWELL, A.E.R.E. (ed.), "Large sets of linear equations (proceedings of the Oxford conference of mathematics and its applications held in April 1970)", Academic Press (1971)

[29] TEWARSON, R.P., "Sparse matrices", Academic Press (1973)

[30] WESTLAKE, J.R., "A handbook of numerical matrix inversion and solution of linear equations", John Wiley & Sons (1968)

[31] COLLINS, R.J., "Bandwidth reduction by automatic renumbering", Int. J. num. Meth. Engng., 6, 345-356 (1973)

[32] ZIENKIEWICZ, O.C. and PHILIPS, D.V., "An automatic mesh generation scheme for plane and curved surfaces by isoparametric coordinates", Int. J. Num. Meth. Engng., 3, 519-528 (1971)

[33] PRENTER, P.M., "Splines and variational methods", John Wiley & Sons (1975)

[34] DUROCHER, L.L. and GASPER, A., "A versatile two-dimensional mesh generator with automatic bandwidth reduction", Comput. & Struct., 10, 561-575 (1979)

[35] LEONHARDT, F., "Prestressed Concrete - Design and Construction", Wilhelm Ernst & Sohn (1964)

[36] LIN, T.Y., "Design of prestressed concrete structures", John Wiley & Sons, New York, (1955)

[37] CATCHICK, B.K., "Prestress analysis for continuous beams: some developments in the equivalent load method", The Struct. Eng., No. 2, Vol. 56B, June 1978, pp. 29-36.

[38] CATCHICK, B.K., "Prestress analysis for continuous beams: some developments in the equivalent load method - Addendum", The Struct. Eng., No. 4, Vol. 56B, December 1979, pp. 82-85.

[39] NIELSEN, M.P., "Yield conditions for reinforced concrete shells in the membrane state", Non-classical Shell Problems: IASS Symposium, Warsaw, 1963. Amsterdam, North Holland Publishing Co., 1964, pp. 1030-1040.

[40] WOOD, R.H., "The reinforcement of slabs in accordance with a pre-determined field of moments", Concrete, Vol. 2, No. 2, February 1968, pp. 69-76.

[41] ARMER, G.S.T., Contribution to the discussion on reference 40, Concrete, Vol. 2, No. 8, August 1968, pp. 319-320.

[42] MORLEY, C.T., "Optimum reinforcement of concrete slab elements against combinations of moments and membrane forces", Magazine of Concrete Research, Vol. 22, No. 72, September 1970, pp. 155-162.

[43] CLARK, L.A. and WEST, R., "The behaviour of solid skew slab bridges under longitudinal prestress", Technical Report, Cement and Concrete Association, December 1974.

[44] MARTI, P. and KONG, K., "Response of reinforced concrete slab elements to torsion", J. of Struct. Engng., Vol. 113, No. 5, May 1987.

[45] THE CONCRETE SOCIETY, "The design of post-tensioned flat slabs in buildings (Recommendations of The Concrete Society Working Party on post-tensioned flat slab construction)", 1974

[46] THE CONCRETE SOCIETY, "Flat slabs in post-tensioned concrete with particular regard to the use of unbonded tendons - design recommendations (Report of a Working Party)", Technical Report No. 17, July 1979

[47] THE CONCRETE SOCIETY, "Post-tensioned flat-slab design Handbook", Technical Report No. 25, 1984

[48] CLARK, L.A., "Concrete bridges design to BS 5400", Longman Inc., New York, (1983)

[49] CP 110 - Code of Practice for the Structural use of Concrete - British Standards Institution (1972)

[50] SABS 0100 - South African Bureau of Standards Code of Practice for the Structural use of Concrete (1980)

[51] ROBINSON, J., "Element evaluation. A set of assessment points and standard test", Proc. F.E.M. in the commercial Environment, Vol. 1, October 1978, pp. 217-248.

[52] BATHE, K.J. and DVORKIN, E.N., "A formulation of general shell elements - The use of mixed interpolation of tensorial components", Int. J. num. Meth. Engng., 22, 697-722 (1986).

APPENDIX A

Courses completed in partial fulfilment of the MSc (Eng) Degree.

THE FOLLOWING POST-GRADUATE COURSES WERE COMPLETED
IN PARTIAL FULFILMENT OF THE
MSc (ENG) DEGREE

AT THE UNIVERSITY OF PRETORIA :

Course		Date Credited	Credit Value
5SR1	Elastisiteitsteorie	1979	
5SR2	Raamanalise	1979	
5SR3	Struktuurdinamika	1979	
5SR5	Numeriese Struktuuranalise	1979	
5SS1	Gewapende Beton	1979	
5SS2	Gewapende Beton	1979	
5TA	Linieere Stelsels	1979	
5TB	Numeriese Metodes	1979	
		Sub total	12

AT THE UNIVERSITY OF CAPE TOWN :

Course		Date Credited	Credit Value
CE 5B12	Computer Aided Design of Structures	1984	3
CE 5B1	Prestressed concrete	1984	5
		Sub total	8
		Total	20

Course Credits : 20
Thesis Credits : 20
Total 40

Total credit requirement for MSc (Eng) Degree : 40

A brief description of the courses are given as follows :

CE 5B12 - COMPUTER AIDED DESIGN OF STRUCTURES

Modelling of structures using finite element analysis methods. Application of computer-aided design techniques to finite element data preparation. Computer programming of the finite element method. Optimum solution techniques for the analysis and design of structures. Computer hardware requirements for computer-aided design.

CE 5B1 - PRESTRESSED CONCRETE

Limit state design of prestressed concrete. Bending, shear and torsion in prestressed structures. Design of continuous structures and composite construction. Partial prestressing. Recent developments in prestressed concrete.

5SR1 - ELASTISITEITSTEORIE (ELASTICITY)

Two- amd three dimensional elasticity. Bending and stability of flat plates. Shear and torsion. Application of finite differences. Plasticity.

5SR2 - RAAMANALISE (FRAME ANALYSIS)

Matrix frame analysis. Dynamic analysis of frame elements and frames. Stability of frame elements and frames.

5SR3 - STRUKTUURDINAMIKA (STRUCTURAL DYNAMICS)

Dynamic analysis of structures with one or more degrees of freedom, with elastic or elasto-plastic response. Continuous parameter systems. Random dynamic loading.

5SR5 - NUMERIESE STRUKTUURANALISE (NUMERICAL STRUCTURAL ANALYSIS)

Introduction to the finite element method. Review of energy methods - Rayleigh-Ritz technique, the displacement method and the force method. Formulation of the finite element method for linear elastic analysis of beams, membranes, axi-symmetric structures and three dimensional structures. Convergence and accuracy. Iso-parametric elements, solution methods and loading conditions.

5SS1 - GEWAPENDE BETON (REINFORCED CONCRETE)

Review of the development of design procedures. Alternative design procedures. Properties of steel and concrete and their influence on design procedures. Determination of ultimate load capacity of sections. Working strength design with particular reference to refinements of the elasticity theory. Ultimate limit states of axial loading, bending moments and combinations. Shear, bond and anchorage. Serviceability limit states (deflection, crack widths and stability).

5SS2 - GEWAPENDE BETON (REINFORCED CONCRETE)

Preliminary design and determination of section sizes (approximate methods and sub-frames). Plastic analysis of structures. Yield lines and patterns. Design for torsion. Strength and ductility of frames (design for seismic loads). Water retaining structures. Functional arrangements of reinforcing. Practical applications and examples.

5TA - LINIEERE STELSELS (LINEAR SYSTEMS)

Systems of normal differential equations. Matrix formulation. Analysis of dynamic response by means of eigenvalues and vectors. Partial differential equations of eliptical, parabolic and hyperbolic form as found in engineering problems. Solution techniques such as separation of variables, Fourier series and the Rayleigh-Ritz method. Non-homogeneous problems and the Duhamel principle.

5TB - NUMERIESE METODES (NUMERICAL METHODS)

Finite difference methods for normal differential equations (boundary value problems and systems of differential equations). Methods for partial differential equations (eliptical, parabolic and hyperbolic forms). Solution of systems of algebraic equations. Methods for linear and interactive techniques. Eigenvalue problems.