A PROGRAM FOR THE AUTOMATIC PLASTIC DESIGN AND ANALYSIS OF PLANE STEEL FRAMES

by

P.D. Griffin

A thesis submitted in partial fulfilment
of the requirements for the degree
Master of Science in Engineering

Department of Civil Engineering
UNIVERSITY OF CAPE TOWN

September 1977

to the memory of my father

Declaration of Candidate

I, Paul Dominic Griffin, hereby declare that this thesis is my own work and that it has not been submitted for a degree at another university.

PGriffin

September 1977

Abstract

The desire for more efficient and realistic structural design of plane steel frames has brought plastic methods of analysis to the fore. Though not commonly used in South Africa, plastic methods have been adopted elsewhere, particularly in the United States. An incremental approach is presented, leading to the determination of the collapse load of a structure and its consequent design. The structure is then analysed using deformation theory analysis, and the results are checked using the A.I.S.C. recommendations, a summary of which is presented. A computer program which automatically executes the design and analysis of any plane steel frame is included, and some of the results obtained from the program are presented.

Acknowledgements

To

Professor J.B. Martin, for his continued interest and encouragement, for his many suggestions and alterations to the manuscript, and for sacrificing his spare time so that this work could be presented in the short time that was available to me.

Colin Dittmer, who had the patience to explain the rudiments of plasticity and computing to me, for his assistance in some of the computational aspects of the thesis and for his major suggestions apropos the final form of this work.

Council for Scientific and Industrial Research and the University of Cape Town for their financial assistance.

Cape Steel Construction Co (Pty) Ltd for their cooperation.

List of Contents

CHAPTER 1

Introduction

1.1 Historical Remarks: [1.1, 1.2, 1.3]

The scientific study of plasticity of metals had its beginnings in 1864 when Tresca published the results of his experiments in punching and extrusion, and postulated a yield condition for the continuum problem. Previous work by Coulomb (1773), Poncelet (1840) and Rankine (1853) was confined mainly to the behaviour of soils.

St Venánt (1870), using the yield condition given by Tresca, introduced constitutive relations for rigid, perfectly plastic materials in plane stress. Levy (1870) generalized this relation to three dimensions. For the next thirty years little significant progress was made.

The first reference to the plastic behaviour of structural members (beams) was made by Ewing (1899)[1.4] who discussed the influence of bending beyond the elastic limit on the distribution of stress and presented stress distributions closely resembling those currently used in simple plastic theory. Ewing further pointed out that the bending moment which will 'break a beam' cannot be calculated by the usual elastic stress formula ($M = \sigma y/I$, where M is the bending moment, σ is the bending stress, I is the moment of inertia of the beam and y is the distance from the neutral axis), because the distribution of stress assumed by that formula ceases to exist as soon as 'overstraining' begins.

It is interesting to note that the elastician A.E.H. Love (1892)[1.5] recognised the undue confidence in elasticity to the neglect of plastic theory by writing:

> 'There is no adequate mathematical theory of set or of after strain, or in fact of any of the phenomena exhibited by materials strained beyond their elastic limit ... Yet it is imperatively necessary that effects which cannot be calculated exactly should be taken into

account in construction, and it is in this sense that elastic theory is at this time behind engineering practice'.

In the theoretical field the most satisfactory advances being made were those by von Mises (1913)[1.16] who introduced an alternative yield condition which together with the flow rule of the St. Venánt-Levy theory provided an acceptable set of constitutive relations.

Between the two wars significant theoretical advances were made by Hencky (1923)[1.17], who interpreted von Mises' purely mathematical approach, Nadai (1923)[1.7], Prandtl (1924)[1.18], Geiringer (1930)[1.19] and Reuss (1932)[1.20], (1933)[1.21]. During this period a great deal of important experimental work was carried out (see, for example, the review paper by Drucker (1956)[1.6]). Von Karmen (1925) analysed the state of stress in rolling, and Siebel and Sachs (1926) put forward results for wire-drawing. Lode (1926) confirmed the Levy-von Mises yield criterion to first order by measuring the deformation of tubes under combined tension and internal pressure. By the early 1930's a theory had been constructed which reproduced the main plastic and elastic properties of an isotropic metal at ordinary temperatures which satisfactorily matched observed results.

In 1931 Nadai gave the first systematic treatment to the theory of plasticity [1.7].

Although the relation between bending moment and curvature of a beam bent into the inelastic range was discussed by Ewing in 1899, it is to Gabor De Kazinczy (1914)[1.8] that credit must go for the first research work illustrating the reserve of plastic strength existing in hyperstatic structures. To him also are due the ideas of the plastic hinge and mechanism of failure.

The fifteen years after 1940 saw the most intensive period of development of basic concepts in classical plastic theory. Hill (1950)[1.1] and Prager and Hodge (1951)[1.9] describe this development comprehensively. Workers in the field of structural plasticity from 1938 included Sir John Baker and his pupils Heyman, Horne, Neal and Roderick, all of Britain, and then, after the war, in the United States mainly at Fritz Laboratory, Lehigh University in Pennsylvania by L.S. Beedle, G.C. Driscoll, T.V. Galambos, R.L. Ketter, T. Kusuda, G.C. Lee, T. Lee, L.W. Lu and B. Thürlimann.

In 1951, simple plastic theory gained its theoretical foundations in the form of the two fundamental theorems of limit analysis, established independently by Greenberg, Drucker and Prager [1.22] in the United states, and by R, Hill [1.1] in Britain. (It was only in 1960 that it was learnt that these theorems had been obtained in 1938 in the Soviet Union by Gvozdev [1.23]). Application of these theorems to beams and framed structures were given by Beedle, Thurlimann and Ketter (1955)[1.10], Heyman (1957)[1.11], Massonnet and Save (1965)[1.12] and Baker and Heyman (1969)[1.13].

In 1953 the European Committee on Concrete, then in 1955, the European Committee for Constructional Steelwork (E.C.C.S.) were set up, and brought powerful support to the acceptance of plastic theory in practical structural problems. The former committee immediately adopted the theory of limit states as the basis of its work, whilst the latter accepted it in 1968.

As early as 1948, the British Standard (BS 449) made provision for plastic design under Cl 29(c) (Fully Rigid Design). The revised 1969 edition states (Cl 9(b) 3):

> 'Alternatively, it (the design) shall be based on the principle of plastic design so as to provide an adequate load factor, and with deflections under working loads not in excess of the limits implied in this British Standard'.

The above paragraph coincides verbatim with the 1948 provision.

Between 1961 and 1962, Spain, India and U.S.S.R. included plastic design in their standard specifications by making general provisions similar to BS 449. In 1963 the United States followed suit. The major drawback for design engineers was the lack of detailed guidelines and consequently the engineer had either to justify his plastic design by personal research, or revert to elastic codes.

In February 1969, the American Institute of Steel Construction introduced detailed specifications for plastic design [1.14]. The code included detailed provisions for beams and girders, one and two storey buildings and a few applications to braced multi-storey frames. This code, with the Canadian recommendation (CSA 516 (1969)) are the most comprehensive in the western world to date. Because of this, the writer has adopted it as a basis for the design examples in this thesis.

In December 1975 the plastic design section of the recommendations of the E.C.C.S. was finalised and adopted. At the time of writing, most industrialized western nations have general provisions, similar to BS 449, included in their specifications.

1.2 Structural Design and Plasticity:

An engineering structure is satisfactorily designed if it can be built to the required economy, and throughout its expected life is able to carry the anticipated loads and perform its intended function.

The design of a steel frame can be based on a number of criteria which constitute a 'limit of structural usefulness'. These are:[1.15]

i) Hypothetical attainment of a specific minimum yield point stress
ii) Attainment of maximum plastic strength
iii) Excessive deflections
iv) Buckling and instability
v) Fatigue
vi) Fracture

Item (i), with item (iv) and (v), formed for many years the basis for design which used the 'allowable stress' concept. Although any of the above six criteria could be used as a basis for 'limit design', the term is usually applied to designs based on item (ii) together with item (iv).

Plastic design embraces primarily item (ii), and is based on the maximum load that a structure will carry as determined from an analysis of strength in the plastic range.

'Allowable-stress design' however is based on an analysis of the structure at working loads with the restriction that the yield stress, divided by an appropriate safety factor, must not be exceeded anywhere in the structure.

1.3 The Load Factor

In steel structures designed by the classical (elastic) methods, maximum stresses under service loads are constrained not to exceed an allowable working stress, given by the yield stress divided by a safety factor. The classical approach thus postulates that the structure is unserviceable if the yield stress divided by a safety factor is exceeded anywhere in the structure.

Plastic analysis, however, assumes that the structure is unserviceable when the applied loads reach a value corresponding to the maximum plastic strength of the structure.

The important consideration in an engineering structure is not whether the yield stress is exceeded at some point, but whether the structure will carry the intended loads or perform its intended function. There is thus really no reason for assuming that the stress in a structure should never exceed the elastic limit. Indeed, in almost all structures, local plastic flow will occur in some points where concentrated loads are applied, and where points of discontinuity in geometry occur. Furthermore, residual stresses as high as half the yield stress may already be present in some elements as a result of the manufacturing process, before loads are ever applied.

The criticism is reinforced when one considers that stresses calculated in the orthodox design method are largely fictitious, because of the assumption that joints and foundations are either rigid or incapable of transmitting bending moment.

Suppose a structure is initially unstressed and that a set of loads W produces a maximum elastic stress σ. Suppose further that the set of loads λW (where λ is greater than unity) produces a maximum elastic stress $\mu\sigma$. The factor λ is termed a load factor, while μ is a stress factor. If the relationship between load and stress is linear throughout the structure up to the load λW, then λ equals μ. This equality does in general not hold for the reasons cited in the previous two paragraphs.

The most important cause of inequality between the factors results from the non-linear stress-strain relationship of steel when the structure is loaded until yield occurs. Plastic theory enables the ratio of the collapse load to a load producing some given extreme fibre stress to be calculated. This collapse load factor is unique to the structural configuration under consideration and to the loads to which that structure is subjected. The concept of a safe stress is by itself misleading and a more general load factor approach is preferred.

Baker, Horne and Heyman[1.2] cite the load factors implicit in BS 449 as being:

i) 1,75 for the most adverse combination of dead plus superimposed load

ii) 1,4 when wind loads are included with the dead plus superimposed loads.

The A.I.S.C. code is less conservative in this respect, and specify (Section 2.1) load factors of 1,7 and 1,3 for dead plus live loads, and dead plus live plus wind loads respectively.

The most advanced concept proposed to date is that of partial safety factors, where the structure is designed to support the external loads after its member strengths have been multiplied by an actual safety factor, which reflects the measure of 'uncertainty' associated with a material. CP 110 for example, which is commonly used for concrete structural design in South Africa, use 1,5 for concrete and 1,15 for steel, but permits these values to go as low as 1,3 and 1,0 respectively (Cl 2.3.3.2).

This concept has been incorporated in the Russian codes for several types of structures (steel, wood, plain concrete, stone, reinforced concrete and prestressed concrete).

Bibliography (Chapter 1)

1.1	R. Hill	1950	The Mathematical Theory of Plasticity. Chapter 1. Oxford.
1.2	J.F. Baker M.R. Horne J. Heyman	1956	The Steel Skeleton. Chapter 1. Cambridge.
1.3	J.B. Martin	1975	Plasticity: Fundamentals and General Results. 3.4 M.I.T. Press.
1.4	J.A. Ewing	1899	The Strength of Materials. Cambridge. Cited in Ref. 1.1, 1.3
1.5	A.E.H. Love	1892	A Treatise on the Mathematical Theory of Elasticity. Cambridge.
1.6	D.C. Drucker	1956	'Stress-strain Relations in the Plastic Range of Metals - Experimentaly and Basic Concepts', Chapter 4 in Rheology, Theory and Applications. Edited by F.R. Eirich, Academic Press (N.Y.). Cited in Ref. 1.3
1.7	A.Nadai	1931	Plasticity. Mc Graw-Hill (N.Y.)
1.8	G. De Kazinczy	1914	'Expériences sur des poutres encastrées' (in Hungarian), Betonszsemle, Vol 2, p 68. Cited in Acier-stahl-steel. 4/1976.
1.9	W. Prager & P.G. Hodge	1951	Theory of Perfectly Plastic Solids, Wiley (N.Y.)
1.10	L.S. Beedle B. Thürlimann R.L. Ketter	1955	Plastic Design in Structural Steel, American Institute of Steel Construction (N.Y.).
1.11	J. Heyman	1957	Plastic Design of Portal Frames, Cambridge.
1.12	C.E. Massonnet M.A. Save	1965	Plastic Analysis and Design, Vol 1, Beams and Frames. Blaisdell (Boston).
1.13	J.F. Baker J. Heyman	1969	Plastic Analysis of Frames. Cambridge.
1.14	American Institute of Steel Construction	1969	Specification for the Design, Fabrication and Erection of Structural Steel for Buildings
1.15	Joint Committee of the Welding Research Council & the American Society of Civil Engineers	1971	Plastic Design in Steel. A Guide and Commentary. A.S.C.E.

1.16	R. von Mises	1913	'Mechanik der festen Köper im plastisch deformablen Zustant', Göttingen Nachr. Math. Phys. Kl., 582. Cited in Ref. 1.3
1.17	H. Hencky	1923	'Veber kinige statisch bestimmte Faelle de Gleichgewichts in plastischen Koerpern', Z. angew. Math. Mech., 3, 241. Cited in Ref. 1.3
1.18	L. Prandtl	1924	'Spannungsverteilung in Plastischen Körpern', Proc. 1st Int. Congr. Appl. Mech., 43, Delft. Cited in Ref. 1.3
1.19	H. Geiringer	1930	'Beit zum Vollständigen ebenen Plastizitätsproblem'. Proc. 3rd Intern. Congr. Appl. Mech. 2, 185. Cited in Ref. 1.3
1.20	A. Reuss	1932	'Fliespotential oder Gleitebenen ?'. ZAMM, 12, 15 Cited in Ref. 1.3
1.21	A. Reuss	1933	'Vereinfachte Berechnung der Plastischen Formänderungsgeschwindigkeiten bei Voraussetzung der Schubspannungsfliessbedingung', ZAMM, 13, 356. Cited in Ref. 1.3
1.22	H.J. Greenberg D.C. Drucker W. Prager	1951	'Extended Limit Design Theorems for Continuous Media', Quart. Appl. Math., 9, 351. Cited in Ref. 1.3
1.23	A.A. Gvozdev	1938	'The Determination of the Value of the Collapse Load for Statically Indeterminate Systems Undergoing Plastic Deformations'. Akad. Nauk. Mowcos-Leningrad, 19-33. Translated by R. Haythornthwaite. Int. J. Mec Sci, 1, 322, 1960. Cited in Ref. 1.3

CHAPTER 2

The Conventional Simple Plastic Design Method

2.1 The General Approach:

The conventional simple plastic design procedure, as applicable to plane frames, is well established (see, for example, Baker, Horne and Heyman,[2.1, 2.2, 2.3] Massonnet and Save,[2.4] and Plastic Design in Steel[2.5]). It has however still to gain general acceptance in South Africa. The procedure will be briefly discussed here.

The geometry of the structure, with boundary conditions, is given, as are the working loads, consisting of dead loads, superimposed loads and wind loads. Loads may act in various combinations leading the designer to choose those combinations which are potentially critical. Load factors are then assigned to each load combination, either according to a chosen specification or at the designer's discretion. The structure is then designed such that flow will not occur when it is subjected to any of the potentially critical factored load cases.

A conventional steel structure is composed of piecewise uniform members. If the section size for each member can be chosen independently, the choice of member sizes involves an optimization process with some design objective. While optimal design procedures are available, they are based on the criterian of least mass, without regard to such practicalities as cost of joints, ready availability of certain section sizes and other practical considerations.

In simple design procedures only one absolute parameter need be chosen. In this work the proportions of the structure is fixed by defining the relative magnitude of the yield moments of the structure's constituent members. The procedure is disadvantageous in that alternative designs may not readily be compared. However, by reproportioning and reanalysing the structure, design alternatives may be examined.

The simple limit analysis approach is conventionally used to determine absolute member sizes. The structure is analysed under each of the potentially critical factored loading conditions, and for each load case the minimum plastic moment for which flow does not occur (or just occurs) is determined (unaccelerated plastic flow). The parameter 'plastic moment' is thus assigned an absolute value for each potentially critical load case, and the largest plastic moment (parameter) is chosen as the criterion for design. It is assumed that flow occurs as a limiting case under the critical load case, but not for others.

Implicit in simple limit analysis is a number of simplifying assumptions. The analysis assumes that:

i) Axial forces and shear forces do not influence the plastic moment of the section

ii) The section is capable of developing the full plastic moment, and

iii) Instability effects are not significant

The classical small displacement assumptions are implied.

The validity of these assumptions must be established in any simple plastic design, and if approached from first principles, require calculations of considerable complexity. Most designers do not have the relevant literature readily available.

A parallel situation occurs in the classical elastic approach. In this case, however, the designer has recourse to the design codes, which, when applied, ensure that the corresponding elastic design assumptions are satisfied.

With regard to plastic design, codes rarely provide the detailed information required. The British Standard BS 449 for instance, which is normally followed in South Africa, gives no recommendations at all, besides requiring that the load factor be adequate, and that deflections be within the limits stipulated by the Standard.

The A.I.S.C. specifications[2.6] contain detailed recommendations for plastic design and have been adopted in this work. The code, adopted in February 1969, is accompanied by a commentary which amplifies its theoretical and philosophical basis.

As the A.I.S.C. code may be unfamiliar to designers accustomed to BS 449, a summary and commentary on its provisions relevant to plastic design is outlined below.

2.2 The A.I.S.C. Recommendations: Plastic Design

The code is divided into two parts. Part 1 deals with specifications for elastic design, and Part 2 exclusively with plastic design, as applicable to one and two storey frames (so called low buildings), beams and girders. It is born in mind that Part 1 is still applicable unless overridden by the provisions of Part 2. Where applicable, units have been metricated. All references below refer to the A.I.S.C. specifications.

2.2.1 Deflections (Section 1.13.1)

As in BS 449, the code requires that the designer pay "due regard" to the deflections produced by the design loads. Beams and girders supporting plastered ceilings shall be so proportioned that the maximum live load deflection does not exceed 1/360 of the span.

2.2.2 The Load Factor (Section 2.1)

Design is on the basis of a "rational analysis" of the structure at ultimate limit state. The structure must be proportioned so as to carry 1,7 times the given live plus dead loads, and 1,3 times those loads acting in conjunction with 1,3 times any specified wind or earthquake loads. The above load factors have been adopted for the present work, although the designer may adopt others at his discretion.

2.2.3 Columns (Section 2.4)

In the plane of bending of columns which would develop a plastic hinge at ultimate loading, the slenderness ratio (ℓ/r) shall not exceed C_c, where

$$C_c = \sqrt{\frac{2\pi^2 E}{\sigma_y}} \qquad (2.1)$$

where E is Young's Modulus (MPa) and σ_y is the yield stress (MPa)

Equation (2.1) is based on elastic column design considerations and implies that the maximum axial stress in a column will not exceed half of the ultimate axial stress for a specified ℓ/r ratio.

Experimental data has shown[2.7, 2.8] that column failures occur at stresses above $\frac{1}{2}P_{cr}$ (Figure 2.1), even when such factors as eccentricity of the column and loading, and residual stresses are taken into account.

The above provision assumes that the column is adequately braced laterally (see Section 2.2.6).

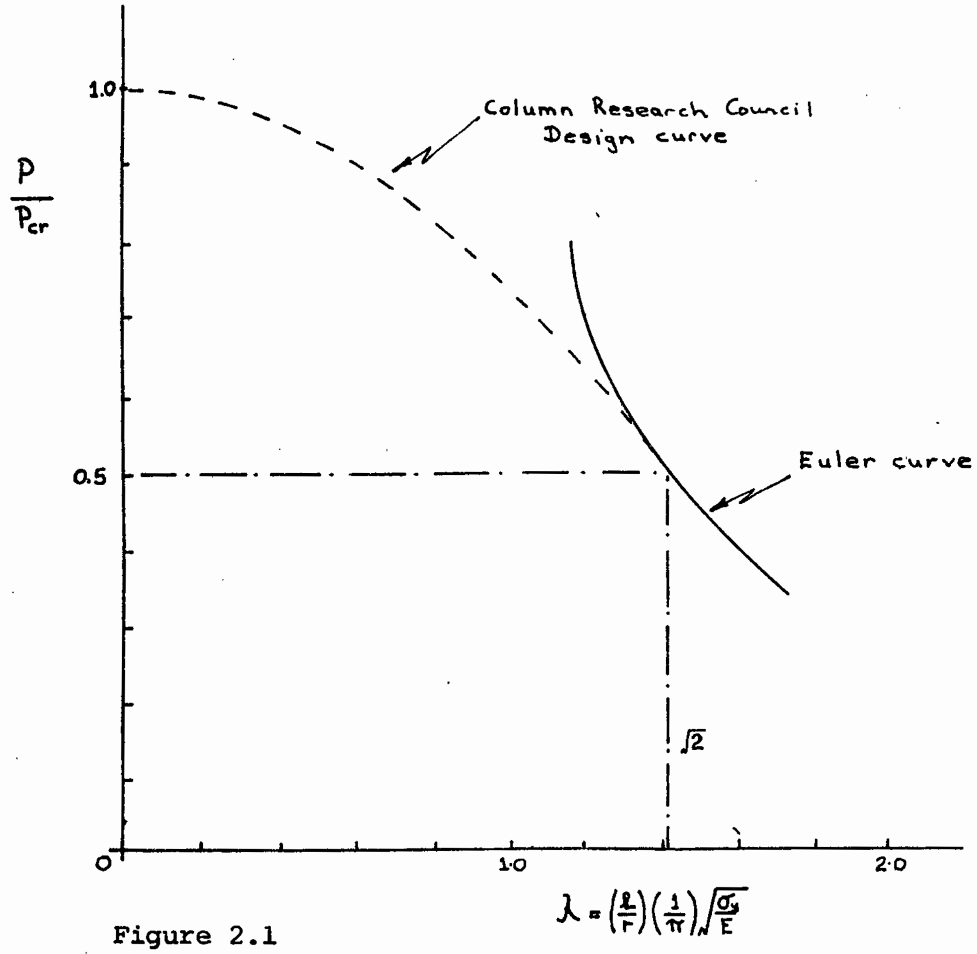


Figure 2.1

For members subjected to axial compression loads only, the maximum force in the member permitted by the code is:

$$P_{cr} = 1{,}7A\sigma_a \qquad (2.2)$$

where A is the gross cross-sectional area of the member, and

$$\sigma_a = \frac{\left[1 - \frac{(\ell/r)^2}{2C_c^2}\right]\sigma_y}{\frac{5}{3} + \frac{3(\ell/r)}{8C_c} - \frac{(\ell/r)^3}{8C_c^3}} \tag{2.3}$$

In this expression ℓ/r is taken as the larger of (ℓ/r_y) or $(k\ell/r_x)$, where
r_y is the radius of gyration about the weak axis
r_x is the radius of gyration about the strong axis
(the web is assumed to be in the plane of the frame)
and $k\ell$ is the effective length of the member

A variable factor of safety has been applied to the estimate of the maximum allowable column compression force in order to obtain permissible working stresses. For very short columns this factor is taken as equal to, or only slightly greater than, that required for members axially loaded in tension ($\sigma_a = 0,6\sigma_y$). This can be justified by the insensitivity of such members to accidental eccentricities.

For longer columns, approaching the limiting slenderness values given by equation (2.1), the factor increases by some 15%. In order to provide a smooth transition between these limits, the factor of safety has been arbitrarily defined by the algebraic equivalent of a quarter sine curve, whose abscissas are the ratio of $k\ell/r$ values to the limiting value C_c (equation (2.1)). The ordinates vary from 5/3 when $k\ell/r = 0$ to 23/12 when $k\ell/r$ equals C_c. The curve is represented by equation (2.3). (see Figure 2.2).

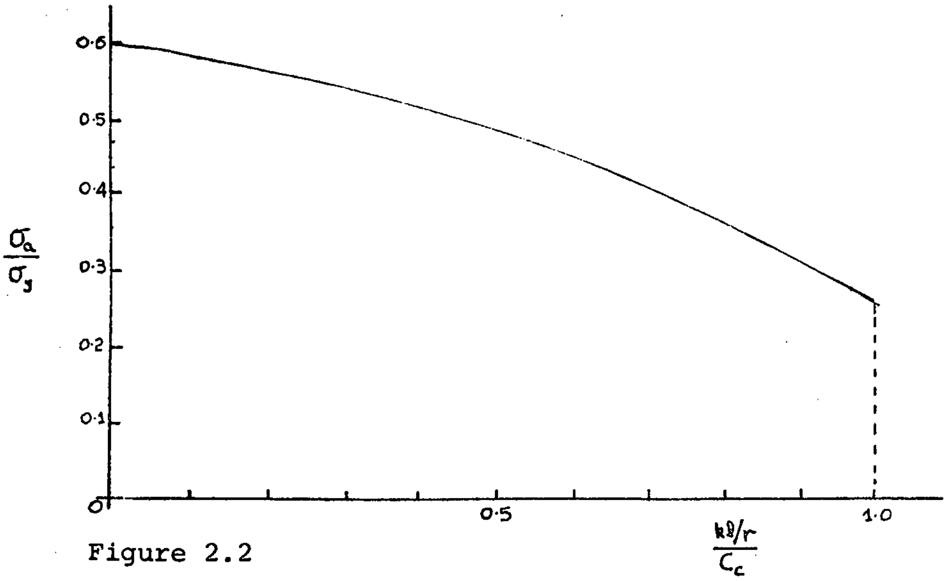


Figure 2.2

Members subject to combined axial loading and bending moment shall be proportioned as so to satisfy the following interaction formulae:

$$\frac{P}{P_{cr}} + \frac{C_m M}{\left(1 - \frac{P}{P_e}\right) M_m} \leq 1{,}0 \qquad (2.4a)$$

and $$\frac{P}{P_y} + \frac{M}{1{,}18M_p} \leq 1{,}0; \quad M \leq M_p \qquad (2.4b)$$

where M is the maximum applied moment

P is the applied axial load

P_e is the Euler load $= \dfrac{\pi^2 EA}{(k\ell/r_x)^2}$

P_y is the maximum axial force under conditions of zero moment, i.e.

$$P_y = \sigma_y . A \qquad (2.5)$$

C_m is defined as follows:

i) For compression members in frames subject to joint translation (sidesway)

$$C_m = 0{,}85$$

ii) For restrained compression members in frames braced against joint translation, and not subject to transverse loading between supports in the plane of bending:

$$C_m = 0{,}6 - 0{,}4 \frac{M_1}{M_2} ; \text{ but not less than } 0{,}4$$

$\frac{M_1}{M_2}$ is the ratio of the smaller to the larger moments at the ends of that portion of the member under consideration which is unbraced in the plane of bending. The ratio is positive for double curvature and negative for single curvature.

iii) For compression members in frames braced against joint translation in the plane of bending and subject to transverse loading between their supports, the value of C_m may be determined by rational analysis. In lieu of such analysis, the code permits either:

a) $C_m = 0{,}85$, for members whose ends are restrained

b) $C_m = 1{,}0$, for members whose ends are unrestrained.

M_m is the maximum moment that can be resisted by a member in the absence of axial load. For columns braced in the weak direction (that is, if the lateral buckling requirements of Section 2.2.6 are met), then

$$M_m = M_p \tag{2.6}$$

For columns unbraced in the weak driection:

$$M_m = \left[1{,}07 - \frac{(\ell/r_y)\sqrt{\sigma_y}}{8\,300} \right] M_p \leq M_p \tag{2.7}$$

where σ_y is the yield stress in MPa

Equation (2.7) was developed empirically on the basis of test observations and provides an estimate of the critical buckling moment, in the absence of axial load, for the case where $M_1/M_2 = -1{,}0$. For other values of M_1/M_2, adjustment is provided by using the appropriate C_m value in equation (2.4a) as defined above. The relevance of equation (2.4a) is amplified below.

The application of moment along the unbraced length of axially loaded members, with its attendant axial displacement in the plane of bending, generates a secondary moment equal to the product of resulting eccentricity and the applied axial load, which is not reflected in the computed bending stress. To provide for this added moment in the design of members subject to combined stresses, equation (2.4a) requires that M be increased by the factor;

$$(1 - \frac{P}{P_e})$$

the terms of which have been defined above.

Depending on the shape of the applied moment diagram, this factor may overestimate the extent of the secondary moment. To accommodate this condition the amplification factor is modified, as required, by a reduction factor C_m.

If the limitation on the slenderness ration outlined in Section 2.2.6 are exceeded, a danger arises in that the rotational capacity of a member may be impaired, due to the combined influence of tortional and lateral deformation, to such an extent that a plastic hinge cannot form. The limiting value of the combined stresses in this case is given by the interaction expression, equation (2.4b), which is represented by Figure 2.3

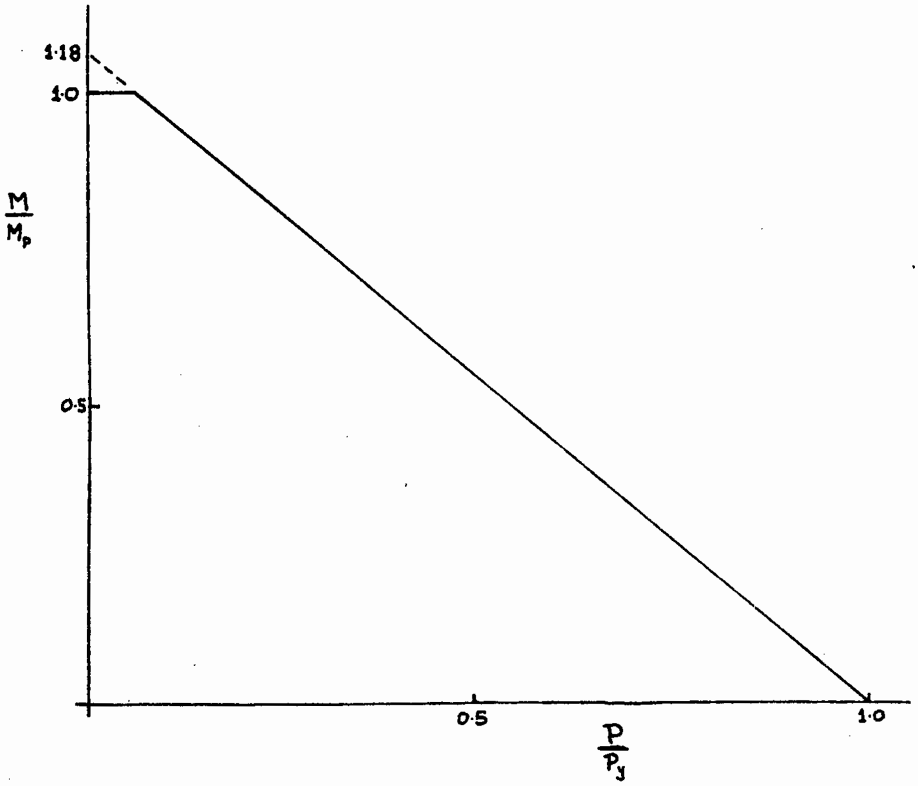


Figure 2.3

If the computed value of M is such that the limitations of equations (2.4a) and (2.4b) are satisfied, then a member is considered to be strong enough to function at a joint where the hinge action is provided by another member entering the joint. An assumed reduction in moment resisting capacity is provided for by using the value M_m (equation (2.7)).

2.2.4 Shear and Web Crippling (Section 2.5, 2.6)

Webs of columns, beams and girders, unless reinforced by diagonal stiffness, shall be proportioned so that:

$$S \leq 0,55\ \sigma_y ad \qquad (2.8)$$

where S is the shear force in Newtons

σ_y is the yield stress in MPa and

a,d are defined in Figure 2.4, and are measured in millimetres

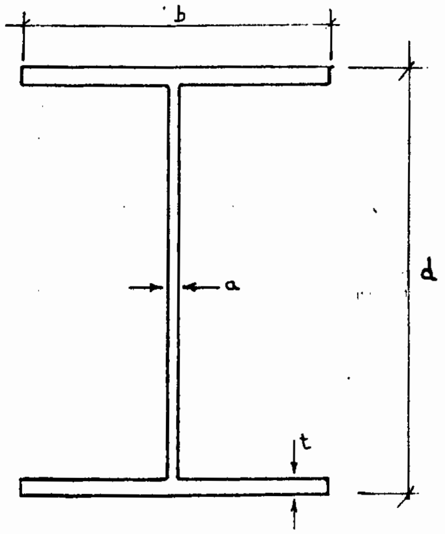


Figure 2.4

Using the von Mises criterion, the average shear stress at which an unreinforced web would fully yield in pure shear is:

$$\tau_y = \frac{\sigma_y}{\sqrt{3}} \qquad (2.9)$$

The plastic bending strength of an I- or H-shaped section is not appreciably reduced until shear yielding occurs over the full effective depth. The full effective depth of a section is defined as the distance between the centroids of its flanges, or 0,95 times the actual depth. Thus

$$S \leq \frac{0,95}{\sqrt{3}} \cdot \sigma_y ad$$

$$\leq 0,55\ \sigma_y ad$$

If the shear force on any section exceeds this amount, the code requires that a pair of diagonal stiffeners be inserted. Alternatively reinforcing plate in contact with the web panel must be welded along its boundaries to the column flanges and/or horizontal stiffeners.

The code emphasises that web stiffeners are required at plastic hinge locations, or points of load application where plastic hinges are expected to form (for example, where large point loads are applied).

The effective ultimate moment of an unstiffened section subject to shear force and bending moment is given by:

$$M_p^{eff} = \sigma_y Z_p \left[1 - \frac{Z_{pw}}{Z_p} \left(1 - \sqrt{1 - \left(\frac{S}{S_p}\right)^2} \right) \right] \qquad (2.10)$$

where Z_p is the plastic section modulus

Z_{pw} is the plastic modulus of the web and

$$Z_{pw} = \frac{a(d - 2t)^2}{4}$$

2.2.5 <u>Minimum Thickness (Width-thickness Ratio)</u> (Section 2.7)

Webs and flanges of I- and H- sections must have sufficient thickness to permit the section to undergo hinge rotations large enough to allow the hinge of the collapse mechanism of a structure to form without reduction of the plastic moment due to local instability. To ensure that this condition is met, the following provisions must be satisfied:

(a) Flanges

The width to thickness ratio for flanges subject to compression involving hinge rotations under limit conditions shall not exceed the values given in Table 2.1, below:

σ_y (Yield Stress in MPa)	$\frac{b}{t}$
250	17
290	16
320	14,8
345	14
380	13,2

Table 2.1

Any section which does not fulfil these requirements may not be used for plastic design.

(b) Webs

The depth to thickness ratio of webs, d/a, is restricted as follows:

i) For sections subject to axial compression alone:

$$\frac{d}{a} \leq \frac{680}{\sqrt{\sigma_y}} \qquad (2.11)$$

ii) For sections subject to pure bending:

$$\frac{d}{a} \leq \frac{1\,100}{\sqrt{\sigma_y}} \qquad (2.12)$$

iii) For sections subject to combined bending and axial compression:

$$\frac{d}{a} \leq \frac{1\,100}{\sigma_y} (1 - 1{,}44 \frac{N}{N_p}) \quad \text{for } \frac{N}{N_p} \leq 0{,}265 \qquad (2.13a)$$

$$\frac{d}{a} \leq \frac{680}{\sqrt{\sigma_y}} \quad \text{for } \frac{N}{N_p} > 0{,}265 \qquad (2.13b)$$

In the above expressions:

σ_y is the yield stress in MPa

N is the axial load

N_p is the squash load in axial compression, and

$N_p = \sigma_y A$ where A is the gross cross-sectional area of the member

Considerable research[2.10] has been required in order to define limiting flange and depth to web thickness ratios below which ample plastic hinge rotations could be relied upon without reduction in the elastic moment due to local buckling.

Table 2.1 and equations (2.13a & b) are obtained from observed results, and, with minor adjustments, are identical to the provisions of the 1963 A.I.S.C. code. (The 1963 formulae are merely multiplied by the factor $\sqrt{250/\sigma_y}$, where σ_y is in MPa, in order to cover the accepted range in yield stresses).

2.2.6 Lateral Bracing (Section 2.9)

An I- section, particularly when loaded into the plastic range, may buckle out of the plane of bending.

The code requires that members be adequately braced to resist lateral and torsional displacements at plastic hinge locations. The laterally unsupported distance from such braced hinge locations to similarly braced adjacent points on a member or frame shall not exceed the value determined by equations (2.14a & b).

$$\frac{\ell_{cr}}{r_y} = \frac{9\ 500}{\sigma_y} + 25 \quad \text{when } 1{,}0 > \frac{M}{M_p} > -0{,}5 \qquad (2.14a)$$

and

$$\frac{\ell_{cr}}{r_y} = \frac{9\ 500}{\sigma_y} \quad \text{when } -0{,}5 > \frac{M}{M_p} > -1{,}0 \qquad (2.14b)$$

r_y is the radius of gyration of the section about its weak axis

σ_y is the yield stress in MPa

M is the lesser of the moments at the ends of the unbraced length

and M/M_p is the end moment ratio, and is positive when the segment is bent in double curvature, and negative when in single curvature

The foregoing provisions do not apply in the region of the last hinge to form in a failure mechanism, nor in members oriented with their weak axis normal to the plane of bending. The latter provision is not applicable in this work.

In the region of the last hinge to form, and in regions not adjacent to plastic hinges, the maximum distance between points of lateral support is governed by elastic requirements.

The A.I.S.C. elastic limits for the unbraced length are defined as the lesser of:

$$\ell_{cr} = \frac{0,20028.b}{\sqrt{\sigma_y}} \qquad (2.15a)$$

or

$$\ell_{cr} = \frac{5,468\ A_f}{\sigma_y . d} \qquad (2.15b)$$

where b,d are defined by Figure 2.4 and are in millimetres

σ_y is the yield stress in MPa

A_f is the area of one flange in square millimetres

Equations (2.14a & b) are empirical expressions. Equation (2.15a) is derived from consideration of beams subjected to uniform moment and of average strain hardening modulus of steels employed for plastic design.

Equation (2.15b) is based on the resistance to lateral buckling offered by St. Venánt torsion and lateral bending[2.11].

When both requirements of equation (2.15a & b) are met, beams with compact sections are insured against overall lateral instability and local buckling or excessive deformation.

Bibliography (Chapter 2)

2.1	J.F. Baker M.R. Horne J. Heyman	1956	The Steel Skeleton: Vol II, Plastic Behaviour and Design. Cambridge University Press.
2.2	J.F. Baker J. Heyman	1969	Plastic Design of Frames. 1. Fundamentals. Cambridge University Press.
2.3	J. Heyman	1971	Plastic Design of Frames. 2. Applications. Cambridge University Press.
2.4	C.E. Massonnet M.A. Save	1965	Plastic Analysis and Design. Vol 1. Beams and Frames. Blaisdell Publishing Company.
2.5	Joint Committee of the Welding Research Council & the American Society of Civil Engineers	1971	Plastic Design in Steel. A Guide and Commentary. Second Ed. A.S.C.E.
2.6	American Institute of Steel Construction	1969	Specification for the Design, Fabrication and Erection of Structural Steel for Buildings. A.I.S.C.
2.7	A. Chajes	1974	Principles of Structural Stability Theory. Prentice-Hall.
2.8	T.V. Galambos	1968	Structural Members and Frames. Prentice-Hall.
2.9	B.O. Johnson	1966	Guide to Design Criteria for Metal Compression Members. John Wiley & Sons. Cited in Ref. 2.8.
2.10	American Society of Civil Engineers		Plastic Design in Steel, A.S.C.E. Manual of Engineering Practise No 41, Second Ed. Section 6.2 Cited in Ref 2.6
2.11	L. Tall	1974	Structural Steel Design. Second Ed. Edited by L. Tall. Art 7.7 Ronald Press Company. New York

CHAPTER 3

An Automated Design Procedure

3.1 Introduction

The primary task in this work is to present a computer program which automatically carries through the design procedure outlined in Section 2.1. The program requires as input data the geometry of the structure with support conditions, the various potentially critical load cases and their respective load factors, and the relative sizes of the structural members (defined by the relative magnitude of the yield moments of the members).

The program consists of six major blocks:

i) Input data
ii) Plastic analysis
iii) Selection of member sizes
iv) Deformation analysis
v) A.I.S.C. design checks
vi) Output

A brief outline of these major blocks is given below. Input data and output will be fully discussed in Chapter 4 (User's Manual). The design process is automatic in the sense that failure to meet A.I.S.C. design checks leads to the redesign of the inadequate member(s). The structure is then reanalysed and checked.

The interdependence of the blocks is presented by means of a flow chart of the program (Figure 3.1). Alternative modes of operation of the program are described in Section 3.7.

3.2 Plastic Analysis

In order to determine the limit load of a structure, subjected to a particular proportional loading condition, the concept of limit analysis is usually adopted. It is a direct approach whereby the load factor at which flow in the structure will occur is determined. No knowledge of the stress history of the structure is required, nor need the elastic properties of the structural members be known.

The basic approaches to limit analysis are the kinematic theorem, an upper bound approach, and the static theorem, a lower bound approach. (See, for example, Greenberg and Prager (1952)[3.1], Baker, Horne and Heyman (1956)[3.2], Hodge (1959)[3.3], Massonnet and Save (1965)[3.4], Prager (1972)[3.5] and Martin (1975)[3.6]).

Various hand methods have been developed for the solution of the limit analysis problem. The kinematic limit theorem is normally applied in the form of the method of combination of mechanisms (Neal and Symonds (1951) (1952)[3.7, 3.8]). This approach is not suited to automation due to the large degree of subjective judgement required of the analyst. Systematic methods have been formulated by Horne (1954)[3.9] and Heyman (1968)[3.10]. The application of the static theorem usually takes the form of the method of superimposing states of self-stress, developed by Heyman (1959)[3.11] which is a special form of the method of minimum weight design developed by Heyman and Prager (1958)[3.12].

Although the limit analysis problem was recognized as a linear mathematical programming problem as early as the late 1940's, it is only since 1965-1966 that the techniques of mathematical programming have been applied to practical structural problems, and not those purely of academic interest. Cohn, Ghosh and Parimi (1972)[3.13] demonstrated that the limit analysis problem, the design problem and shakedown with simple plastic theory could be formulated as L.P. problems and therefore could be solved in a unified way with standard algorithms.

This numerical approach has a severe drawback when viewed in the context of the overall design procedure adopted here, in that the problem is formulated without reference to shear and axial forces, which necessitates a reformulation when elastic or elastic-plastic analysis is required in Block (iv), Section 3.1.

In order to overcome these difficulties inherent in the approaches to the solution of the limit analysis problem previously mentioned, an incremental approach is adopted. Loads are increased in a stepwise fashion and the resulting changes in deformation are computed. The limit load is then the load for which flow is found to occur. Viewed in isolation, this approach may be less efficient than a direct limit analysis, but in the context of the program it is considered to be the best analysis procedure.

Elastic section properties are required for the incremental analysis but are not known since members have not been chosen at this point. The limit load is however independent of the elastic properties and it is consequently adequate to assign arbitrary, but sensible, numerical values to the axial rigidity and flexural stiffnesses of the members.

An arbitrary value of the full plastic moment is assigned to one section, and the full plastic moments of other sections are accordingly assigned to the remaining members of the structure to give the selected structure proportion.

The limit load factor (that is, the factor by which the working loads are multiplied to cause flow) is determined for each potentially critical load case.

The theoretical approach to the incremental problem is described in detail in Appendix A. It is based on a theorem by Martin [3.6], and a solution algorithm suggested for the analysis of a plane truss by Martin and Reddy (1975)[3.14].

3.3 Selection of Member Sizes

The incremental analysis provides the designer with a collapse load factor associated with each potentially critical load case. The critical load case has associated with it the smallest of these collapse load factors. The piecewise uniform members of the structure have up to now been defined by their respective full plastic moments, the numerical value of which were arbitrarily assigned, but according to the preselected proportions of the structure.

The structure may now be 'scaled up' to real proportions. The actual full plastic moment required by a member of the structure at collapse is calculated as follows: The product of the proportional plastic moment of that member and the load factor associated with that load case is determined and then divided by the smallest collapse load factor.

The quotient of the full plastic moment (obtained above) divided by the yield stress gives the plastic modulus of the section required. It then becomes a simple sorting procedure to choose the section whose plastic modulus is just greater than that required.

This procedure is carried out for each member. The sorting procedure is easily automated. For the purposes of this work all the I-sections listed in the abridged version of the Handbook on Hot Rolled Structural Steel Sections [3.15] were sorted in ascending order of the magnitude of their plastic moduli. The computer program, having calculated the plastic modulus required, scans the list of plastic moduli and selects that plastic modulus just greater than the one required. That section is then chosen as a member of the structure.

The designer may choose to substitute his own set of sections (for example H-sections, and selected I-sections) from which the choice of section may be made. This is dealt with more fully in Chapter 4, User's Manual.

3.4 Deformation Analysis

The structure, now composed of members whose section properties are known, must be reanalysed under working and factored loads in order to determine the deformations and internal forces of the structure.

The problem is no longer one of limit analysis but an elastic or elastic-plastic analysis problem with known loading conditions being applied to the structure.

Martin and Reddy [3.16] cite an algorithm for deformation analysis, which yields a solution to the elastic or elastic-plastic analysis problem. The method gives both elastic and plastic deformation in the structure, as well as internal forces. Deformation theory is dealt with in more detail in Appendix B.

Although incremental theory might readily be reapplied in order to analyse the structure, the writer believes deformation theory is advantageous in that:

i) It is a 'one-step' solution procedure, and thus converges to the solution quicker than the incremental approach.
ii) It is computationally more efficient than incremental theory.
iii) It provides a check on the incremental analysis (and vice versa).

It should be noted that incremental theory must be used as the first analysis procedure because the problem was one of limit analysis, in order to obtain acceptable structural members. Deformation theory analysis cannot be used effectively to find the load at which flow occurs.

The program analyses the structure using deformation theory for all loading conditions (not only under the critical load case) to ensure that deflection requirements are met both at working and factored loads under all circumstances.

The results of the analysis of the structure under critical factored loading are then checked according to the A.I.S.C. specifications.

3.5 The Automated Application of the A.I.S.C. Design Checks

Discussion of the computer application of the A.I.S.C. design check will follow the sequence adopted in Section 2.2. Any references given refer to the A.I.S.C. code [2.6].

3.5.1 Deflections (Section 1.13.1)

The computer program has a facility whereby a limiting deflection magnitude may be placed on any node of the structure. The program will then automatically compare the deflection at that node both under working loads and factored loads to the stipulated permissible deflection. If the deflection is excessive, the program will comment, 'EXCESSIVE DEFLECTION AT NODE: X' and the analysis procedure will stop.

The designer may then re-examine the structure and decide which are the most appropriate members to stiffen, and then, with the new members included, reanalyse the structure.

3.5.2 The Load Factor (Section 2.1)

Load factors corresponding to the various potentially critical load cases are read in as data. In the examples presented in Chapter 5 the A.I.S.C. recommendations for load factors have been adopted.

3.5.3 Columns (Section 2.4)

The calculated value of C_c (equation (2.1)) is compared with the slenderness ratio of each column. If the slenderness ratio of the column exceeds C_c, redesign of that member is necessary. The radius of gyration required by the member is given by the present radius of gyration multiplied by the factor $(\ell/r)/C_c$. The new member is thus chosen by finding that section which has a radius of gyration just greater than the new radius of gyration calculated above. The maximum axial compressive load P_{cr} (equation (2.2)) is compared to the axial force P in the member. If this value is exceeded the gross sectional area of the member is increased by the factor P/P_{cr}, and a new member is chosen whose cross-sectional area just exceeds the increased section. This check is applied to columns only.

Equations (2.4a & b) are applied to columns and beam-columns. The value of C_m (equation (2.4a)) is input as data. (See Chapter 4, User's Manual). M_m (equation (2.7) in equation (2.4a)) is calculated automatically. If the requirements of either equation (2.4a or b) are not met, the program prints: 'INTERACTION FORMULA ONE (OR TWO) UNACCEPTABLE', and the program stops after completion of the remaining design checks. No redesign process is included because of the many alternatives open to the designer when this situation arises. He may increase the cross-sectional area of the member, increase r_{xx} or decrease r_{yy}. The change in section is not a linear adjustment as in other redesign procedures. The remaining checks are completed in order to present the designer with the performance of other members of the structure when subjected to the remaining design checks.

3.5.4 Shear and Web Crippling Section 2.5, 2.6

The maximum allowable shear force S (equation (2.8)) is compared to the shear force in each member. If S is exceeded, the program prints: 'DIAGONAL STIFFENERS REQUIRED', as the code stipulates. If the effective ultimate moment M_p^{eff} of an unstiffened section subject to shear force and bending moment (equation (2.10)) is less than 98% of the yield moment of the section, the program prints: 'EFFECTIVE YIELD MOMENT UNACCEPTABLE'. The structure is then reanalysed using the effective yield moment instead of the actual yield moment of that member.

3.5.5 Minimum Thickness (Width-thickness Ratio) Section 2.7

This clause is included for completeness for, in general, sections not suited for plastic design would not be considered in the first instance. The restrictions put on (i) $\frac{b}{t}$ ratios in Table 2.1, and (ii) $\frac{d}{a}$ ratios in equations (2.11), (2.12), (2.13a & b).

i) If such sections are inadvertantly considered, the program prints 'ELEMENT X IS NOT COMPACT AND MAY NOT BE USED FOR PLASTIC DESIGN', where x is the number of the element. After all other checks have been completed the process will stop.

ii) Similarly the $\frac{d}{a}$ ratio is checked, according to equations (2.11), (2.12), and (2.13a & b). If the requirements of these equations are not met, the program prints: '$\frac{d}{a}$ RATIO FOR SPECIFIED $\frac{N}{N_p}$ RATIO UNACCEPTABLE' and then 'ELEMENT X IS NOT COMPACT AND MAY NOT BE USED FOR PLASTIC DESIGN'.

3.5.6 Lateral Bracing Section 2.9

The critical unbraced length ℓ_{cr} in equations (2.14a & b) are calculated and printed for all points in the structure where a plastic hinge has formed.

At all other points the elastic provisions concerning lateral bracing are applied (equation (2.15a & b)). Since lateral bracing requirements are a function only of the section properties of the structural members, the applicable critical length defined by either equation (2.15a or 2.15b) is printed only once if the structure is uniform. The designer may then fulfil those minimum bracing requirements by bracing the structure appropriately.

At positions of plastic hinges two values of the critical length are given, for single and double curvature. By consulting the relevant bending moment diagram, the designer is able to decide on the applicable bracing requirements.

3.6 A Comment on Output of Design Check Results

Because of the volume of printout required to present each of the above stipulations in detail, the writer has included options in the program whereby the user may choose either a tabulated result of the design checks, or a full listing (which includes both the minimum requirements of the code, and the value supplied by the design, together with comments).

For the reason cited above, only the tabulated version will be presented in the examples (Chapter 5). This consists of 'OK', 'NO' or 'N/A' responses to the various design checks if they respectively fulfil the requirements of the code, do not fulfil the requirements of the code, or if that design check is not applicable to a certain member.

If a member fails to meet a code requirement, the cause will be elaborated on by presenting the relevant minimum requirement, together with the value supplied by the design, with comments.

If all the aforegoing provisions (Section 3.5) are met, the program will print: 'ALL DESIGN CHECKS SUCCESSFULLY COMPLETED', and the program stops.

3.7 The Program as an Analysis and Design Checking Tool

Besides being used to design a structure, the program may alternatively be used purely for the analysis and checking of an existing or proposed design.

The user inputs section sizes and various working load cases, with their respective load factors. The structure is analysed for each load case using deformation theory, and the results of these analyses are checked, as outlined in Section 3.2. All analyses are checked because the critical load case will not is general be known. Access to this facility is discussed in Chapter 4 (User's Manual, Data Input).

The analysis is automatically terminated if the user inputs inadequate sections which result in a collapse mechanism forming. He may then increase his section sizes and submit the problem for reanalysis.

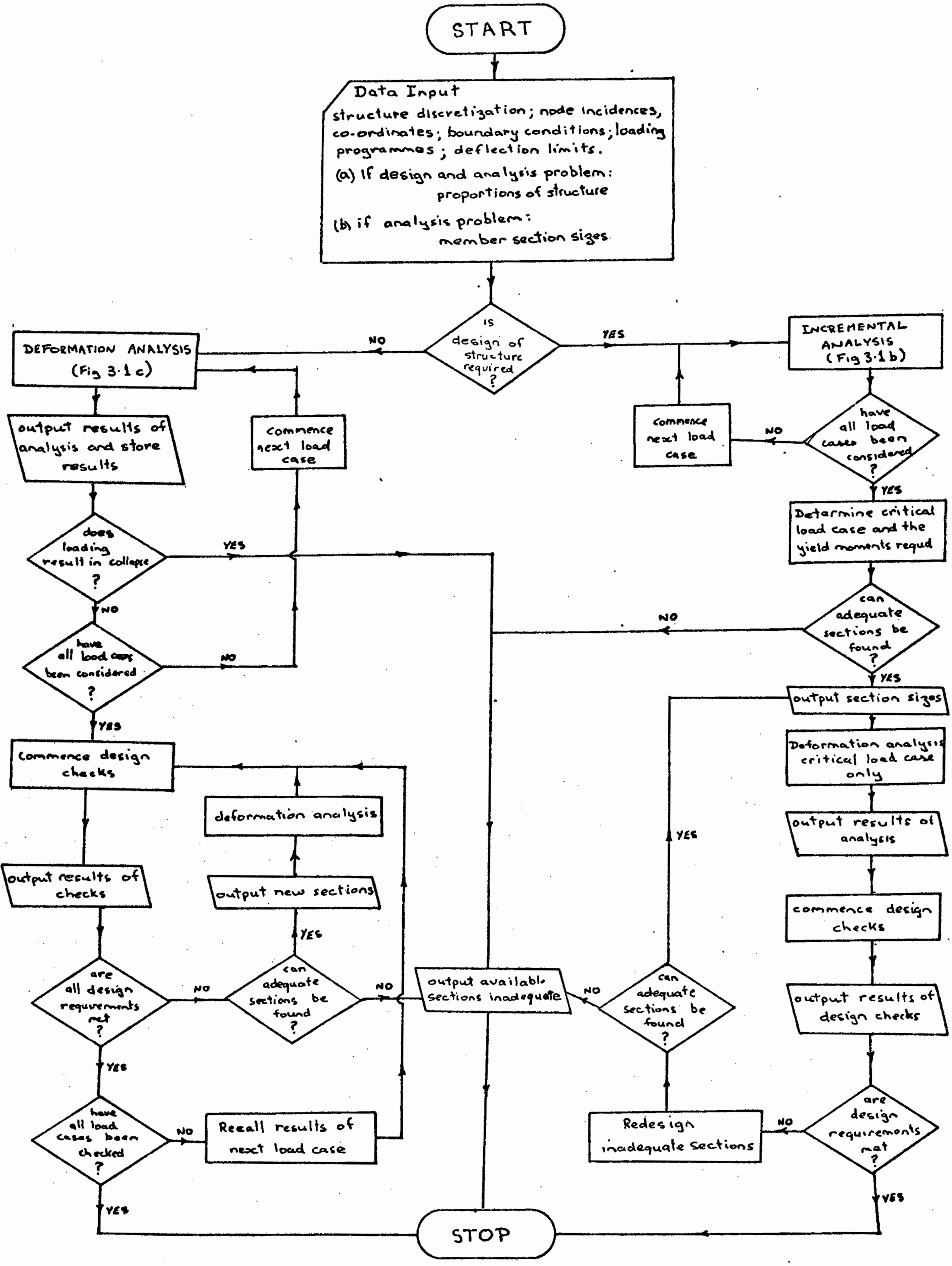

START
Data Input
structure discretization; node incidences, co-ordinates; boundary conditions; loading programmes; deflection limits.
(a) If design and analysis problem: proportions of structure
(b) if analysis problem: member section sizes.
is design of structure required ?
NO
YES
DEFORMATION ANALYSIS (Fig 3.1 c)
INCREMENTAL ANALYSIS (Fig 3.1 b)
output results of analysis and store results
commence next load case
does loading result in collapse ?
YES
NO
have all load cases been considered ?
NO
YES
commence design checks
deformation analysis
output results of checks
output new sections
YES
are all design requirements met ?
NO
can adequate sections be found ?
NO
YES
have all load cases been checked ?
NO
Recall results of next load case
YES
STOP
commence next load case
NO
have all load cases been considered ?
YES
Determine critical load case and the yield moments requd
can adequate sections be found ?
NO
YES
output section sizes
Deformation analysis critical load case only
output results of analysis
commence design checks
output results of design checks
are design requirements met ?
NO
YES
Redesign inadequate Sections
can adequate sections be found ?
YES
NO
output available sections inadequate

Figure 3.1(a)

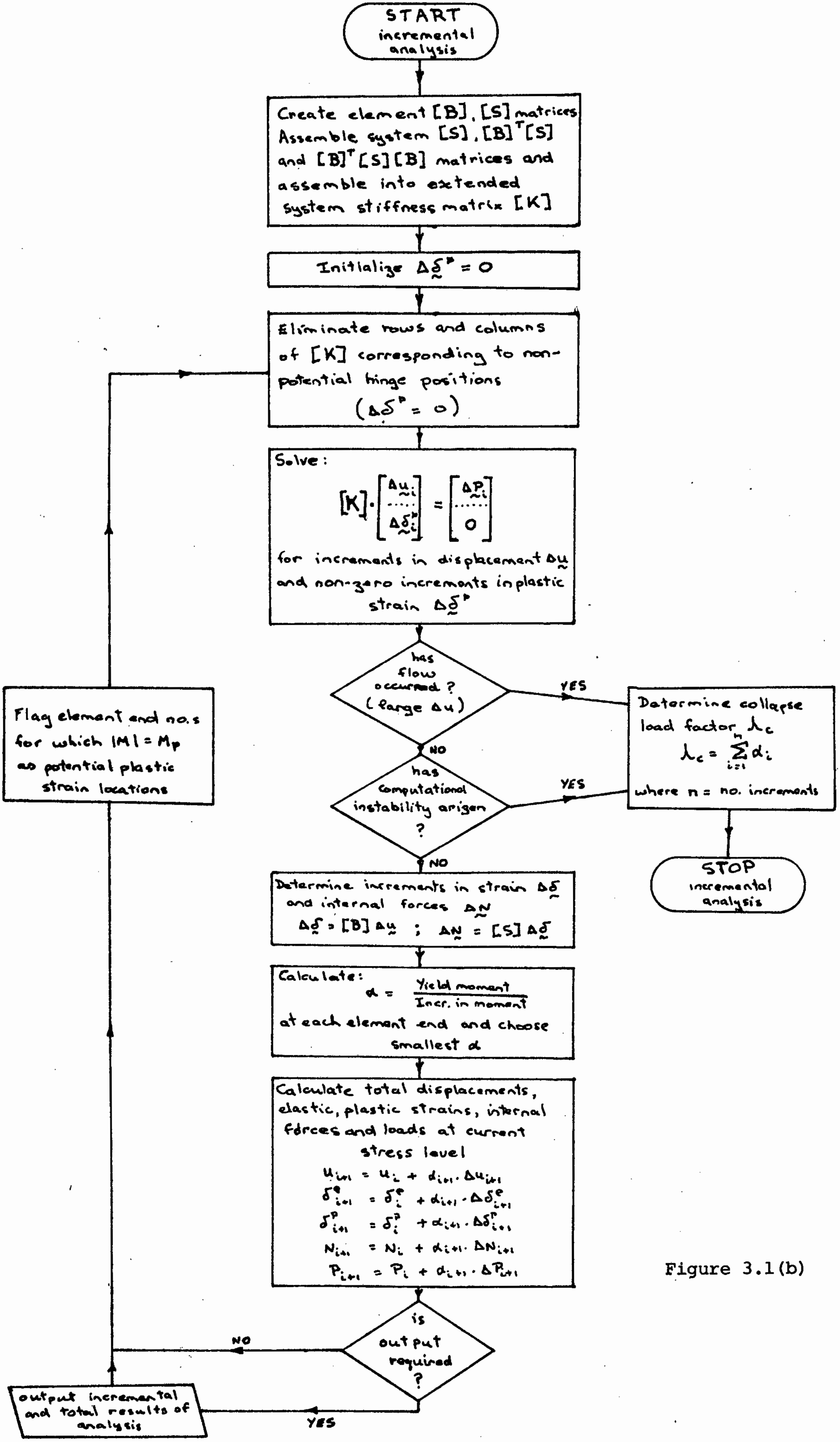

START incremental analysis
Create element [B], [S] matrices Assemble system [S], $[B]^T[S]$ and $[B]^T[S][B]$ matrices and assemble into extended system stiffness matrix [K]
Initialize $\Delta\underset{\sim}{\delta}^p = 0$
Eliminate rows and columns of [K] corresponding to non-potential hinge positions ($\Delta\delta^p = 0$)
Solve:
$[K]\cdot\begin{bmatrix}\Delta\underset{\sim}{u}_i \\ \Delta\underset{\sim}{\delta}^p_i\end{bmatrix} = \begin{bmatrix}\Delta\underset{\sim}{P}_i \\ 0\end{bmatrix}$
for increments in displacement $\Delta\underset{\sim}{u}$ and non-zero increments in plastic strain $\Delta\underset{\sim}{\delta}^p$
has flow occurred ? (large Δu)
YES
NO
Determine collapse load factor λ_c
$\lambda_c = \sum_{i=1}^{n} \alpha_i$
where n = no. increments
has computational instability arisen ?
YES
NO
STOP incremental analysis
Determine increments in strain $\Delta\underset{\sim}{\delta}$ and internal forces $\Delta\underset{\sim}{N}$
$\Delta\underset{\sim}{\delta} = [B]\,\Delta\underset{\sim}{u}$; $\Delta\underset{\sim}{N} = [S]\,\Delta\underset{\sim}{\delta}$
Calculate: $\alpha = \frac{\text{Yield moment}}{\text{Incr. in moment}}$ at each element end and choose smallest α
Calculate total displacements, elastic, plastic strains, internal forces and loads at current stress level
$u_{i+1} = u_i + \alpha_{i+1}\cdot\Delta u_{i+1}$
$\delta^e_{i+1} = \delta^e_i + \alpha_{i+1}\cdot\Delta\delta^e_{i+1}$
$\delta^p_{i+1} = \delta^p_i + \alpha_{i+1}\cdot\Delta\delta^p_{i+1}$
$N_{i+1} = N_i + \alpha_{i+1}\cdot\Delta N_{i+1}$
$P_{i+1} = P_i + \alpha_{i+1}\cdot\Delta P_{i+1}$
is output required ?
NO
YES
output incremental and total results of analysis
Flag element and no.s for which |M| = M_p as potential plastic strain locations

Figure 3.1(b)

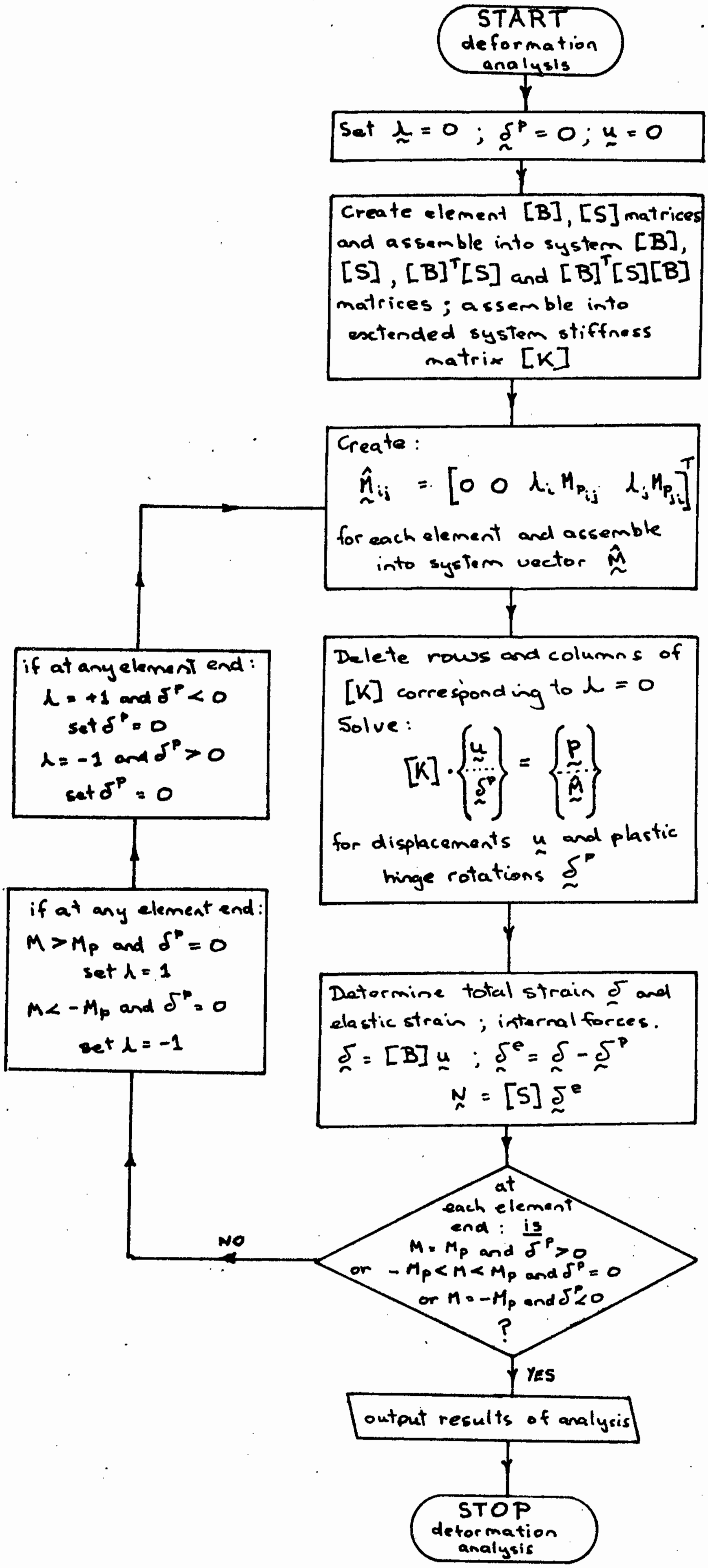

START deformation analysis
Set $\lambda = 0$; $\delta^P = 0$; $u = 0$
Create element [B], [S] matrices and assemble into system [B], [S], $[B]^T[S]$ and $[B]^T[S][B]$ matrices ; assemble into extended system stiffness matrix [K]
Create : $\hat{M}_{ij} = [0 \; 0 \; \lambda_i M_{P_{ij}} \; \lambda_j M_{P_{ji}}]^T$ for each element and assemble into system vector $\hat{M}$
Delete rows and columns of [K] corresponding to $\lambda = 0$
Solve : $[K] \cdot \begin{Bmatrix} u \\ \delta^P \end{Bmatrix} = \begin{Bmatrix} P \\ \hat{M} \end{Bmatrix}$ for displacements u and plastic hinge rotations δ^P
Determine total strain δ and elastic strain ; internal forces.
$\delta = [B] u$; $\delta^e = \delta - \delta^P$
$N = [S] \delta^e$
at each element end : is $M = M_p$ and $\delta^P > 0$ or $-M_p < M < M_p$ and $\delta^P = 0$ or $M = -M_p$ and $\delta^P < 0$?
NO
YES
if at any element end: $M > M_p$ and $\delta^P = 0$ set $\lambda = 1$; $M < -M_p$ and $\delta^P = 0$ set $\lambda = -1$
if at any element end: $\lambda = +1$ and $\delta^P < 0$ set $\delta^P = 0$; $\lambda = -1$ and $\delta^P > 0$ set $\delta^P = 0$
output results of analysis
STOP deformation analysis

Figure 3.1(c)

Bibliography (Chapter 3)

3.1	H.J. Greenberg W. Prager	1951	'Limit Design of Beams and Frames. Proc. Amer. Soc. Civil Engineers, 77, Separate No 59.
3.2	J.F. Baker M.R. Horne J. Heyman	1956	The Steel Skeleton, Vol 2. Cambridge.
3.3	P.G. Hodge	1959	Plastic Analysis of Structures, McGraw-Hill (N.Y.)
3.4	C.E. Massonnet M.A. Save	1965	Plastic Analysis and Design. Vol 1. Beams and Frames. Blaisdell Publishing Company. (N.Y.)
3.5	W. Prager	1972	'Limit Analysis: The Development of a Concept'. Foundation of Plasticity. (ed. A. Sawczuk), Noordhoff, Ch. 2.
3.6	J.B. Martin	1975	Plasticity: Fundamentals and General Results. M.I.T. Press.
3.7	B.G. Neal P.S. Symonds	1951	' The Calculation of Collapse Loads for Framed Structures', Proc. I.C.E., 35, 21.
3.8	B.G. Neal P.S. Symonds	1952	'The Rapid Calculation of Plastic Collapse Loads for a Framed Structure', Proc. I.C.E., 1, (3), 58.
3.9	M.R. Horne	1954	'A Moment-distribution Method for the Analysis and design of Structures by the Plastic Theory', Proc. I.C.E., 3.
3.10	J. Heyman	1968	'Bending Moment distributions in Collapsing Frames', in Engineering Plasticity (ed. J. Heyman and F.A. Leckie), Cambridge.
3.11	J. Heyman	1958	'Automatic Plastic Analysis of Steel Framed Structures under Fixed and Varying Loads', Rep. IBM 20382, Div Appl. Math., Brown Univ., Providence, R.I.
3.12	J. Heyman W. Prager	1958	'Automatic Minimum Weight Design of Steel Frames', J. Franklin Inst., 266: 339-364.
3.13	M.Z. Cohn S.K. Ghosh S.R. Parimi	1972	'Unified Approach to the Theory of Plastic Structures', J. of Eng. Mech. Div. Proc. Am. Soc. Civil Engrs. EM 5,1133-1158

3.14	J.B. Martin B.D. Reddy	1975	'A Programming Approach to the Solution of the Rate Problem in Elastic, Plastic Solids'. Tech. Report 751. Structural Mechanics Group. Dept of Civil Engineering, UCT.
3.15	The South African Rolled Steel Producers' Coordinating Council	1976	'Abridged Version of Handbook on Hot Rolled Structural Steel Sections'.
3.16	J.B. Martin B.D. Reddy J. Baggett	1977	'Elastic-Plastic Deformations in Plane Frames'. The Civil Engineer in South Africa. Vol 19, no 5.

CHAPTER 4

ADPF User's Manual

4.1 General Description

ADPF (Automatic Design of Plane Frames) is a finite element computer program capable of automatic plastic design and analysis of plane steel frames consisting of piecewise uniform members, and subjected to any number of static loading conditions. The program checks the results of the analysis according to the A.I.S.C. specifications and is capable of redesigning any member which does not meet the requirements of that code. A further facility of the program is its ability to analyse and check an existing design.

The workings of the program are presented by means of a macro flow chart (Figure 3.1(a)), the contents of which are amplified by Figures 3.1(b) and 3.1(c).

Numerical results obtained from the program are presented and discussed in Chapter 5.

A detailed description of the internal logic of the program will not be presented here. ADPF is written in modular form, that is, it consists of a main program and a series of independent subroutines. The function of the main program is to call relevant subroutines which perform the actual analysis and input/output. The subroutines may act alone or in conjunction with others, each subroutine having a specific function.

Among the general features of the program are the following:

a) All arithmetic is performed in single precision.

b) Certain printed output is optional. The user may select exactly what output he requires by entering the relevant option in the data input (Section 4.3.3 (1)).

c) By means of an option, the user may request just the analysis of an existing or proposed structure, instead of the entire design procedure.

d) Computer storage allocation can be adjusted according to the size of the problem by changing the dimensions of certain arrays at the head of each subroutine (Section 4.5)

e) The program is capable of analysing a structure for a number of load cases in one run.

f) All input is in free format.

A detailed description of data input with sample data (Table 4.1) is given in Section 4.3.3. Included in Section 4.2 is a typical runstream, illustrating access and execution of ADPF.

The program is written in FORTRAN V as implemented on the UNIVAC 1106 computer. It is maintained permanently both on disc and magnetic tape at the University of Cape Town Computer Centre, and is therefore readily available to any authorised interested user.

In the descriptions that follow (except for data input), it is assumed that the user has a basic knowledge of FORTRAN.

4.2 A Typical Runstream

The following runstream is used to access and execute ADPF.

```
@RUN,/Bb̸RUNID,ACCNT.NO/USERID,PROJECTID,TIME,PAGES
@ASG,AXb̸PLASLM*ADPF.
@ASG,AXb̸PLASLM*SECTN.
@ASG,Tb̸13
@ASG,Tb̸14
@ASG,Tb̸15
@USEb̸12,PLASLM*SECTN.
@XQTb̸PLASLM*ADPF.ABS

              DATA  INPUT

@FIN
```

The program must be executed in batch mode, due to the size of the output and core storage requirements.

4.3 Data Input

The format of data input for the design, analysis and checking problem, and the analysis and checking problem, is identical. Dummy data values are input where not required for the problem under consideration. This approach saves further data preparation if, for example, the user wishes to examine the same problem purely for analysis purposes.

4.3.1 Units

All input and output data have the dimensions of kilonewtons, metres or radians, unless otherwise stated (Section 4.3.3)

4.3.2 Sign convention

Node coordinates are defined in a global coordinate system. The sign convention adopted is illustrated in Figure 4.1

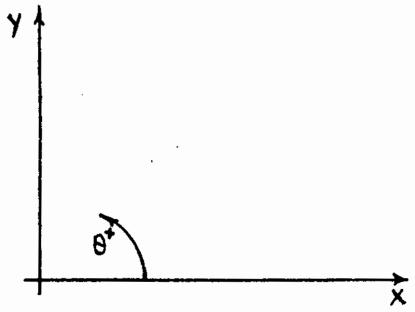


Figure 4.1

Bending moments in the computer printout are in a local coordinate system (Figure 4.2).

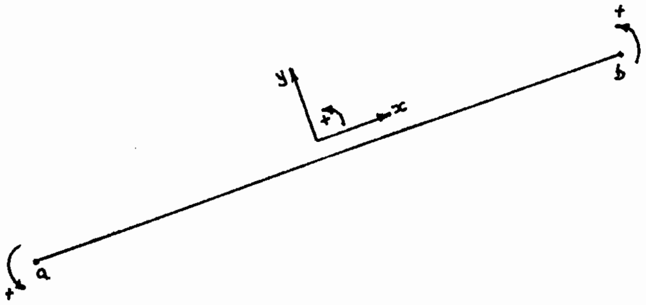


Figure 4.2

4.3.3 Detail of Data Input

1. Parameter Statement

NROWP,NCOLP,NSEC,IBP,ISHORT,ITAB

where:

NROWP (integer)	number of rows in the extended system stiffness matrix as stored in the computer (Section 4.3.4(a)).
NCOLP (integer)	number of columns in the extended system stiffness matrix as stored in the computer (Section 4.3.4(a)).
NSEC (integer)	total number of sections being considered for selection as stored in the file SECTN (Section 4.4).
IBP (integer)	IBP = 0 for the design, analysis and checking problem IBP = 1 for the analysis and checking problem
ISHORT (integer)	ISHORT = 0 for a comprehensive printout of results of all iterations in the incremental and deformation analyses. ISHORT = 1 for an abridged output of results, which includes results of analyses at both working and ultimate loads, and sections chosen in the design.
ITAB (integer)	ITAB = 0 for a comprehensive printout of design check results. ITAB = 1, for a condensed presentation of design check results. If a section proves inadequate, the reason will be detailed.

It is suggested that the option 1,1 be used for ISHORT, ITAB, as this gives all the information required and avoids unnecessarily bulky output.

2. Structure Statement

NE,NN,NLC,E,AE,EI,YSTRS

where:

NE (integer)	total number of elements
NN (integer)	total number of nodes
NLC (integer)	total number of load cases to which the structure is being subjected
E (integer)	Young's Modulus in Pascals

AE (real)	Young's Modulus*Area of Section in kN (see Section 4.3.4(b))
EI (real)	Young's Modulus*2nd Moment of Area of Section in kNm^2 (see Section 4.3.4(b))
YSTRS (real)	Yield stress of steel in Pascals

3. Load Factors

RLAMDA(1),RLAMDA(2),. . . . .,RLAMDA(NLC), (real), the load factors associated respectively with each load case.

4. Effective Length Factor and Buckling Restraint Parameter

EFFLEN(1),CM1(1),EFFLEN(2),CM1(2),. . .,EFFLEN(NE),CM1(NE), (real) respectively the effective length factor and a counter to indicate the degree of buckling restraint (see Section 4.3.4(c)), for the ordered elements.

5. Columns Statement

NC,IC(1),IC(2),. . .,IC(NC)

where:

NC (integer)	the total number of columns in the structure. (For the purposes of the program, a column is defined as a member which is expected to be subjected to significant axial loading. A distinction between columns and beam-columns is necessary when the AISC specifications are applied).
IC(1),IC(2),. . ., IC(NC) (integers)	element numbers of the members which are to be treated as columns. Any element not included in this set is assumed to behave as a beam.

6. Element Incidences

NBEAM(1,1),NBEAM(1,2),NBEAM(2,1),NBEAM(2,2),...,NBEAM(NE,1),NBEAM(NE,2)

where:

NBEAM(1,1)	node number of 'a' end of element 1.
(integer)	(see Figure 4.2)
NBEAM(1,2)	node number of 'b' end of element 1.
(integer)	(see Figure 4.2)
⋮	
NBEAM(NE,1)	node number of 'a' end of element NE
(integer)	
NBEAM(NE,2)	node number of 'b' end of element NE
(integer)	

7. Relative Yield Moments

MP(1),MP(2),...,MP(NE) (real), relative yield moments of ordered elements (see Section 4.3.4(d)).

8. Nodal Coordinates

COORDX(1),COORDY(1),COORDX(2),COORDY(2),...,COORDX(NN),COORDY(NN) (real), Global x and y coordinates of ordered nodes (metres)

9. Boundary Conditions

$$\left\{\begin{matrix} N_i, N_i \\ I,J,K \end{matrix}\right\} \quad \cdots\cdots \quad *$$

$$\left\{\begin{matrix} \quad \\ \quad \end{matrix}\right\}$$

$$\vdots$$

$$\left\{\begin{matrix} \quad \\ \quad \end{matrix}\right\}$$

- 1, -1

where N_i,N_i (integers) is the number of the node, repeated, which has boundary conditions I,J,K (integers) and where

$I = 0$ - if constrained in global x-direction,
$I = 1$ - if unconstrained in global x-direction,
$J = 0$ - if constrained in global y-direction,
$J = 1$ - if unconstrained in global y-direction,
$K = 0$ - if constrained against rotation in x-y plane
$K = 1$ - if unconstrained against rotation in x-y plane

Totally unconstrained nodes need not appear under the boundary conditions. A boundary condition subgroup such as * above corresponds to each node which has any degree of constraint. However, should consecutive node numbers $N_i,N_{i+1},..N_k$ have the same boundary conditions I,J,K this can be input as

$$\left\{\begin{matrix} N_i,N_k \\ I,J,K \end{matrix}\right\}.$$

- 1, -1 indicates the end of the boundary conditions.

10, 11. Load Cases

The input for each load case has the same format:

$$\left.\begin{matrix} K \\ \left\{\begin{matrix} N_i,N_i \\ P_x,P_y,M \end{matrix}\right\} \; \ldots\; * \\ \left\{\begin{matrix} \\ \\ \end{matrix}\right\} \\ \vdots \\ \left\{\begin{matrix} \\ \\ \end{matrix}\right\} \\ -1,-1 \end{matrix}\right\} \;\ldots\ldots\ldots\; **$$

where K (integer) is the number of the load case; N_i,N_i (integers) is the number of the node, repeated, which carries external loads P_x,P_y and M (reals), and where

P_x is load applied in global x-direction,
P_y is load applied in global y-direction, and
M is moment applied in x-y plane, anticlockwise positive. Nodes which have no external loads need not appear in a load case. A nodal load subgroup such as * above corresponds to each node which has any external load. However should consecutive node number $N_i, N_{i+1}, \ldots, N_k$ have the same loads P_x, P_y and M, these can be input as

N_i, N_k
P_x, P_y, M

- 1, - 1 indicates the end of the load case.

A load case data group such as ** above corresponds to each load case. Thus there must be NLC such groups (see Structure Statement).

12. End of Loading Statement

- 1 signifies the end of all loading input

13. Deflection Checks at Working Load

$\{N_i, I_i, D_i\}$ . . . *

{ }
|
|
|
{ }

- 1

where	N_i (integer)	number of a node at which a deflection is to be checked
	I_i (integer)	I = 1 for deflection check in global x-drection, I = 2 for deflection check in global y-direction, or I = 3 for rotation check
and	D_i (real)	limiting value of displacement (metres or radians) at working load

A data subgroup * must be entered for each deflection check.
- 1 indicates the end of deflection checks at working load. If there are no deflection checks at working load - 1 must be entered.

14. Deflection Checks at Ultimate Load

The form of the input is the same as that for deflection checks at working load (see § 13), where D_i is now the limiting value of displacement (metres or radians) at ultimate load. - 1 indicates the end of deflection checks at ultimate load. If there are no deflection checks at ultimate load - 1 must be entered.

15. Analysis Problem Only

For the analysis and checking problem alone (i.e. no design), the sectional properties of each member are required. The input is of the form

$$\left\{\begin{array}{l} N_i, N_i \\ SP(1), SP(2), \ldots, SP(20) \end{array}\right\} \ldots *$$

$$\left\{\begin{array}{l} \qquad\qquad\qquad\qquad \\ \\ \end{array}\right\}$$

$$\vdots$$

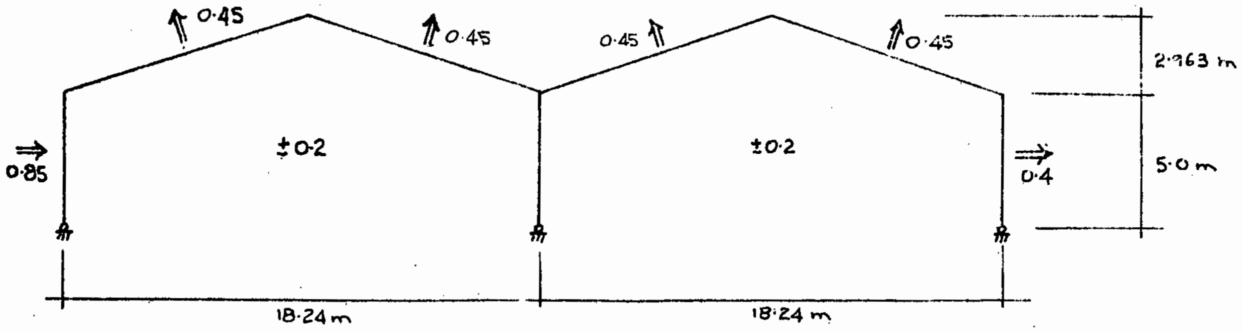
$$\left\{\begin{array}{l} \qquad\qquad\qquad\qquad \\ \\ \end{array}\right\}$$

where N_i, N_i (integers) is the number of the element, repeated, which has sectional properties $SP(1), SP(2), \ldots, SP(20)$, (reals). (see Section 4.4). A data subgroup * must be given for each element of the structure. However, should consecutive element number $N_i, N_{i+1}, \ldots, N_k$ have the same sectional properties $SP(1), SP(2), \ldots, SP(20)$, these can be entered as

$$\left\{\begin{array}{l} N_i, N_k \\ SP(1), SP(2), \ldots, SP(20) \end{array}\right\} .$$

- 1 signifies the end of sectional properties.

Figure 4.3 Two-bay Pinned Portal Frame - Design and Analysis Required

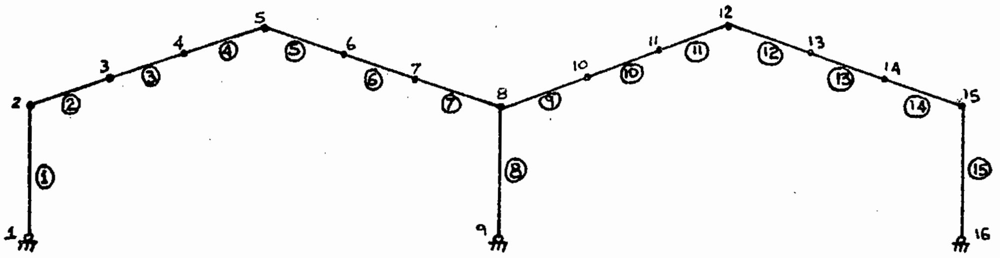


Loading: Dead Load = 1,42 kN/m uniformly distributed

Live Load = 2,45 kN/m uniformly distributed

Wind Load: q = 3,13 kN/m. Pressure coefficients are shown in Figure 4.3

Figure 4.4 Element and Node Numbering



ⓧ : denotes the element number x.

x : denotes the node number x.

UNIVERSITY OF CAPE TOWN COMPUTER CENTRE

FORTRAN Coding Form

PROGRAM PLASLM*ADPF

PROGRAMMER P.D. GRIFFIN

DATE 18/9/77

PUNCHING INSTRUCTIONS

GRAPHIC

PUNCH

PAGE 1 OF 3

CARD ELECTRO NUMBER

STATEMENT NUMBER

FORTRAN STATEMENT

IDENTIFICATION SEQUENCE

Table 4·1

```
1.  108,70,40,0,1,1
2.  15,16,2,200.E9,2.E6,4.E4,250.E6
3.  1.7,1.3
4.  1.5,1.,1.,1.,1.,1.,1.,1.,1.,1.,1.,1.,1.,1.,1.5,1.
    1.,1.,1.,1.,1.,1.,1.,1.,1.,1.,1.,1.,1.5,1.
5.  3,1,8,15
                                                  18.24,5.0
6.  1,2,2,3,3,4,4,5,5,6,6,7,7,8,8,9,8,10,10,11,11,12,13,13,14,14,15,15,16
7.  100.,100.,100.,100.,100.,100.,100.,100.,100.,100.,100.,100.,100.,100.,100.
8.  0.,0.,0.,5.,3.04,6.,6.08,7.,9.12,7.963,12.16,7.,15.2,6.,18.24,0.,21.28
    6.,24.32,7.,27.36,7.963,30.4,7.,33.44,6.,36.48,5.,36.48,0.
9.  1,1
    0,0,1
    9,9
    0,0,1
    16,16
    0,0,1
    -1,-1
10  1
    2,2
    0.,-5.88,0.
    3,7
    0.,-11.78,0.
    8,8
    0.,-5.88,0.
```

PD UCT

UNIVERSITY OF CAPE TOWN COMPUTER CENTRE

FORTRAN Coding Form

PROGRAM	PLASLM *ADPF		PUNCHING INSTRUCTIONS	GRAPHIC	PAGE 2 OF 3
PROGRAMMER	P. D. GRIFFIN	DATE 18/9/77		PUNCH	CARD ELECTRO NUMBER*

Table 4.1 (cont.)

STATEMENT NUMBER	FORTRAN STATEMENT	IDENTIFICATION SEQUENCE

```
    10,14
    0.,-11.76,0.
10  15,15
    0.,-5.88,0.
    -1,-1
    2
    2,2
    13.73,-5.88,0.
    3,7
    0.,-11.76,0.
11  8,8
    0.,-5.88,0.
    10,14
    0.,-11.76,0.
    15,15
    2.08,-5.88,0.
    -1,-1
12. -1
    2,1,0.15
    5,2,0.15
13  12,1,0.15
    15,1,0.15
    -1
14. 2,1,0.3
```

PD UCT

UNIVERSITY OF CAPE TOWN COMPUTER CENTRE

FORTRAN Coding Form

PROGRAM PLASLM * AJPF

PROGRAMMER P.D. GRIFFIN

DATE 18/9/77

PUNCHING INSTRUCTIONS

GRAPHIC

PUNCH

PAGE 3 OF 3

CARD ELECTRO NUMBER*

STATEMENT NUMBER

FORTRAN STATEMENT Table 4.1 (cont.)

IDENTIFICATION SEQUENCE

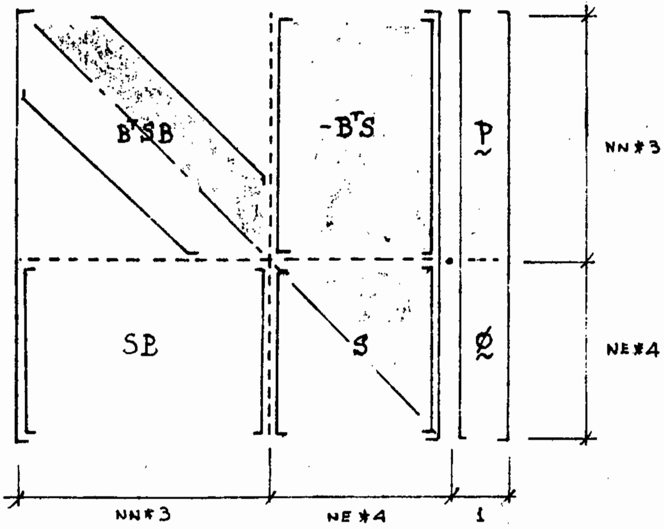
14

```
5,2,0.3
12,1,0.3
15,2,0.3
-1
```

PD UCT

4.3.4 Amplification of Data Input

a) The extended system stiffness matrix [K] is square and symmetric. The elastic system stiffness matrix B^TSB, a submatrix of [K], is tightly banded (see Figure 4.3). It is thus not necessary to store the entire matrix as this entails duplication of information, as well as storage of zero values. In the program only part of the upper triangle is stored, as shown in Figure 4.5.



is stored as

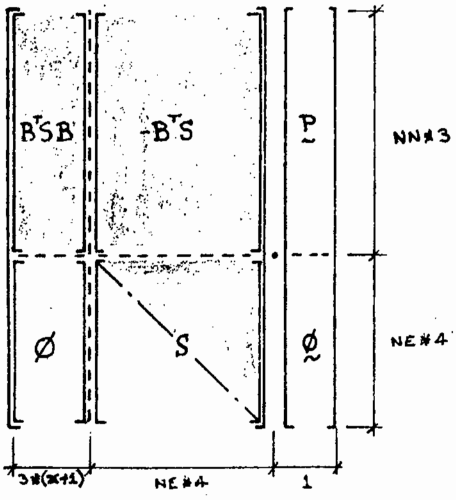


Figure 4.5

where NE is the number of elements,

NN is the number of nodes,

and x is the greatest numerical difference in end node numbers for all elements.

Thus, total number of rows = NROWP = NN*3+NE*4,

and total number of columns = NCOLP = 3*(x+1)+NE*4+1.

b) In the initial stage of the design process, actual section sizes are not knonw. Arbitrary, but sensible values must be assigned to EI and AE for the incremental analysis. Values of the order of 10^4 kN and 10^6 kNm^2 for EI and AE respectively are applicable. For the analysis and checking problem however, the actual sectional properties of the members are input as data. Consequently dummy values of EI and AE can be entered in the Structure Statement, for example, 0.,0.

c) The designer must decide on the effective length factor k, where kℓ is the effective length of a member and ℓ is the actual length of the member, for each element of the structure. This information is required for buckling checks in the design checking phase of the program. Although the effective lengths of members defined as beams are not used in the program, dummy values must be entered. The parameter C_m is defined in Section 2.2.3 according to four categories of restraint conditions. The designer must stipulate either:

CM1(I) = 1., for compression members in frames subject to joint translation,

CM1(I) = 2., for restrained compression members in frames braced against joint translation, and not subject to transverse loading between supports in the plane of bending,

CM1(I) = 3., for members whose ends are restrained, or

CM1(I) = 4., for members whose ends are unrestrained,

where I is the number of the element.

d) In the design process, the proportions of the structure must be defined by stipulating the relative magnitude of the element yield moments, in the order of element numbering. A datum value of 100 has been chosen. Thus, any one element can be assigned a yield moment of 100, and the yield moments of each of the remaining elements must be given relative to this datum value.

4.4 Section Properties Data File

The user may choose any table H and/or I sections from which he wishes the design to be selected. The sectional properties are stored in the file SECTN. as follows:

The section sizes, with all their properties are listed as they appear in The Abridged Version of Handbook on Hot Rolled Structural Steel Section[3.15], and are ordered in ascending magnitude of $Z_{p\ell}$ (xx). The dimensions of the section properties remain as in the handbook. For example the section

Serial Size	P_t	d	b	a	t	r	h_1	A	I_{xx}	Ze_{xx}	Z_{pl_x}	r_{xx}	I_{yy}	Ze_{yy}	Z_{pl_y}	r_{yy}	C	d/t
mm	kg/m	mm	mm	mm	mm	mm	mm	$10^{-3}m^2$	$10^{-6}m^4$	$10^{-6}m^3$	$10^{-6}m^3$	mm	$10^{-6}m^4$	$10^{-6}m^3$	$10^{-6}m^3$	mm	$10^{-9}m^4$	
305 x 165	46·1	307·1	165·7	6·7	11·8	8·9	266	5·878	99·35	647	721·5	130	8·957	108·1	165·5	39	288·2	26

is entered as:

305., 165., 46.1, 307.1, 165.7, 6.7, 11.8, 8.9, 266., 5.878, 99.35,

647., 721.5, 130., 8.957, 108.1, 165.5, 39., 288.2, 26.

At the time of writing SECTN. contains all the compact I-sections listed in the Handbook $^{3.15}$. The list of sectional properties stored in the file SECTN. can be altered.

a) To delete a section from the file:
Locate the section size in the file using the edit processor of the computer, then delete the relevant lines (there are twenty two numerical entries per section, which are in free format and occupy two lines of data).

b) To insert new sections into the file:
Since the sections are ordered in ascending order of $Z_{p\ell}(xx)$, the position of the new section is dictated by the magnitude of its plastic modulus. Once its position has been found (again using the edit processor), the section size and properties may be entered, as outlined in paragraph 2 of this section.

Note that the number of sections in the file SECTN. must be entered in the Parameter Statement (see § 1). At the time of writing the file contains data corresponding to forty different I-sections, therefore NSEC = 40. For those not familiar with methods of editing files, it is suggested that an experienced person be consulted before any changes are attempted.

4.5 Expansion of Storage Allocation for Larger Problems

All the arrays used in ADPF appear in labelled common blocks at the head of each subroutine (see Appendix C, Program Listing). At present the program is dimensioned to accommodate a structure with a maximum of twenty three elements, twenty four nodes, and ten load cases.

To investigate larger problems the storage capacity of the program must be altered by increasing the dimensions of various arrays in the common blocks, as follows:

i) COMMON/EXT/R_K(157,98), appears at the head of each subroutine and corresponds to the storage assignment for the extended system stiffness matrix. The values 157,98 correspond to NROWP, NCOLP (see Section 4.3.4(a)). If a larger problem requires more storage than is presently allocated, the above two parameters must be increased to the required magnitude, as outlined in Section 4.3.4(a).

ii) Ten other labelled common blocks exist, called COMMON/BLK1/, COMMON/BLK2/....COMMON/BLK10/. At present each array in the ten labelled COMMON blocks contains the numbers 23 or 73 as dimension limits. These correspond, respectively, to the maximum number of elements, and (NN*3+1), where NN is the maximum number of nodes.

Should a particular problem require increased storage facilities, the relative dimension limits (23 and 73) must be changed so as to accommodate the larger problem. Note that if the dimension limits are to be altered, this must be done throughout every COMMON statement of every subroutine. The names of subroutines are listed below.

If any changes are made to the program, the program must be recompiled and mapped as follows:

```
@ADD0PLASLM*ADPF.FOR
@ADD0PLASLM*ADPF.MAP
```

The above two elements, FOR and MAP, contain the following:

```
(i)   FOR                      (ii)   MAP
@FOR ADPF.MAIN                 @PACK ADPF.
  "    "  .DATA                @PREP ADPF.
  "    "  .SOLVE               @MAP,IN ADPF.ABS
  "    "  .ELTMAT              IN ADPF.MAIN
  "    "  .STRSTR              "    "  .DATA
  "    "  .OUTPUT              "    "  .SOLVE
  "    "  .ITER                "    "  .ELTMAT
  "    "  .SOLVED              "    "  .STRSTR
  "    "  .ELTMAD              "    "  .OUTPUT
  "    "  .STRSTD              "    "  .ITER
  "    "  .OUTPUD              "    "  .SOLVED
  "    "  .ITED                "    "  .ELTMAD
  "    "  .SORT                "    "  .STRSTD
  "    "  .CONTRL              "    "  .OUTPUD
  "    "  .CHECK               "    "  .ITED
  "    "  .NEWSEC              "    "  .SORT
                               "    "  .CONTRL
                               "    "  .CHECK
                               "    "  .NEWSEC
```

CHAPTER 5

Examples

5.1 Two-bay Pinned Portal Frame

The configuration of the structure with the various loading conditions is presented in Figures 4.3 and 4.4.

The solution of the problem is presented by means of the computer output, which follows. (Page 58)

A 254*146*24.5 section was initially chosen for all members. This section was then found inadequate for the columns of the strucutre and a 356*171*67.2 was substituted, which proved adequate.

5.2 Fixed End Rectangular Portal Frame (uniform)

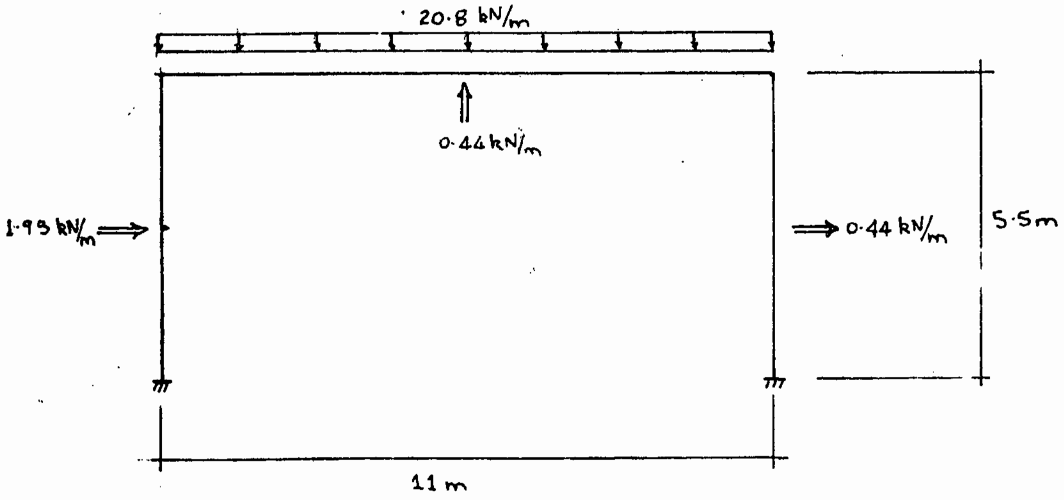


Figure 5.1

Dead and snow load = 20,8 kN/m

Wind load : Figure 5.1

Load case 1 : Dead + snow load

Load factor = 1,85

Load case 2 : Dead + snow + wind load

Load factor = 1,4

The above load factors are used in order to compare the results obtained from ADPF to those of Massonnet and Save [1.12], page 339.

	Section Size	$Z_{p\ell}$
ADPF	356*171*67.2	$1213 \times 10^{-6}\ m^3$
Massonnet and Save	16WF45	$1344 \times 10^{-6}\ m^3$

The 16WF45 has dimensions 409*179. No equivalent metric section exists. The next lower American section has a $Z_{p\ell}$ less than the required value ($1270 \times 10^{-6}\ m^3$). Although the section obtained by ADPF has a marginally smaller $Z_{p\ell}$ than the minimum value cited by Massonnet and Save, it proves adequate.

5.3 Two-bay Rectangular Portal Frame (non uniform)

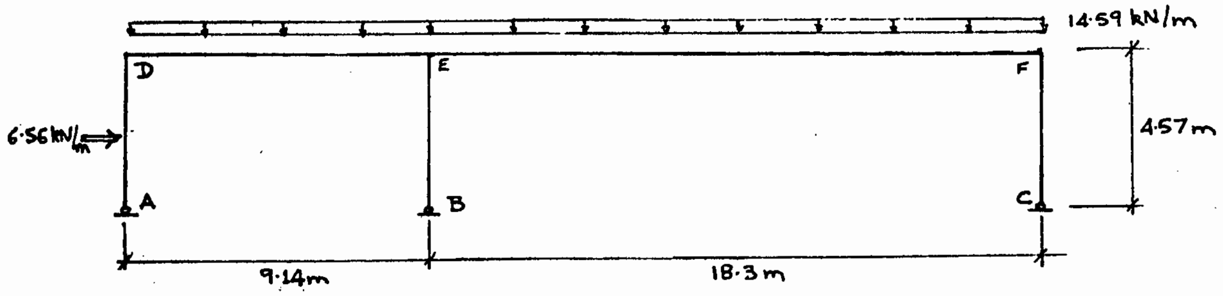


Figure 5.2

Dead Load = 14,59 kN/m

Wind Load = 6,56 kN/m (see Figure 5.2)

Load Case 1 : Dead load

Load factor = 1,85

Load Case 2 : Dead + wind load

Load factor = 1,4

Again, the above load factors are used for comparison purposes (Massonnet and Save $^{1.12}$, page 355).
The proportions of the structure are:

AD,DE : 1
EB : 3
EF,FC : 4

Member	ADPF	Massonnet & Save	Remarks
AD,DE	406*140*46.3	356*171*44.8	1 Section heavier
EB	533*210*109	533*210*92.5	1 Section heavier
EF,FC	610*229*125	610*229*125	Equivalent section

Because there is not a one to one correspondence between the sets of sections from which Massonnet and Save and ADPF choose, equivalent sections are not always expected to be chosen.

```
*********************************
*                               *
* AUTOMATIC DESIGN AND ANALYSIS *
*       OF PLANE FRAMES         *
*                               *
*          P.D.GRIFFIN          *
*  DEPT. OF CIVIL ENGINEERING   *
*   UNIVERSITY OF CAPE TOWN     *
*                               *
*                               *
*       UNIVAC 1106 EXEC8       *
*                               *
*********************************


*********************************
* JOB DESCRIPTION :             *
*                               *
*                               *
* Two bay pinned uniform portal frame *
*                               *
* (i) Dead + superimposed load (λ = 1·7) *
* (ii) Dead + superimposed + wind (λ = 1·3) *
*        Design and analysis    *
*                               *
* DATE : 1 Oct 77.  USER : P.D.GRIFFIN *
*********************************
```

```
                                                    UNITS : KILONEWTONS AND METRES
                                                    ******************************



   NUMBER OF ELEMENTS =    15
   NUMBER OF NODES    =    16

   ELASTIC MODULUS   = .200000+09


          COORDINATES OF NODES

   NODE          X               Y
     1      .000000         .000000
     2      .000000         .500000+01
     3      .304000+01      .600000+01
     4      .608000+01      .700000+01
     5      .912000+01      .795300+01
     6      .121600+02      .700000+01
     7      .152000+02      .600000+01
     8      .182400+02      .500000+01
     9      .182400+02      .000000
    10      .212800+02      .600000+01
    11      .243200+02      .700000+01
    12      .273600+02      .795300+01
    13      .304000+02      .700000+01
    14      .334400+02      .600000+01
    15      .364800+02      .500000+01
    16      .364800+02      .000000


   ELEMENT      NODES              LENGTH
              /--------/

      1         1      2      .500000+01
      2         2      3      .320025+01
      3         3      4      .320025+01
      4         4      5      .318388+01
      5         5      6      .318888+01
      6         6      7      .320025+01
      7         7      8      .320025+01
      8         8      9      .500000+01
      9         9     10      .320025+01
     10        10     11      .320025+01
     11        11     12      .318888+01
     12        12     13      .318888+01
     13        13     14      .320025+01
     14        14     15      .320025+01
     15        15     16      .500000+01


    BOUNDARY CONDITIONS : 0=FIXITY , 1=FREEDOM

    NODE    X    Y    ROTATION
      1     0    0       1
      9     0    0       1
```

```
  16      0      0         .


LOADING PROGRAMME:


LOADING.    1

                         NODE        PX             PY          MOMENT
                            2     .000000     -.588000+01     .000000
                            3     .000000     -.117600+02     .000000
                            4     .000000     -.117600+02     .000000
                            5     .000000     -.117600+02     .000000
                            6     .000000     -.117600+02     .000000
                            7     .000000     -.117600+02     .000000
                            8     .000000     -.588000+01     .000000
                           10     .000000     -.117600+02     .000000
                           11     .000000     -.117600+02     .000000
                           12     .000000     -.117600+02     .000000
                           13     .000000     -.117600+02     .000000
                           14     .000000     -.117600+02     .000000
                           15     .000000     -.588000+01     .000000


LOADING     2


                         NODE        PX             PY          MOMENT
                            2     .109300+02  -.588000+01     .000000
                            3     .000000     -.117600+02     .000000
                            4     .000000     -.117600+02     .000000
                            5     .000000     -.117600+02     .000000
                            6     .000000     -.117600+02     .000000
                            7     .000000     -.117600+02     .000000
                            8     .000000     -.588000+01     .000000
                           10     .000000     -.117600+02     .000000
                           11     .000000     -.117600+02     .000000
                           12     .000000     -.117600+02     .000000
                           13     .000000     -.117600+02     .000000
                           14     .000000     -.117600+02     .000000
                           15     .298000+01  -.588000+01     .000000


              SELF WEIGHT IS AUTOMATICALLY INCLUDED INTO THE LOAD VECTORS BEFORE DEFORMATION ANALYSIS COMMENCES




                                             ********************************
                                             RESULTS OF INCREMENTAL ANALYSIS
                                             ********************************
```

```
                  B
          <------------------>
          ********************    A
          *                  *    !
          *                  *    !
          *********  *********    !
                  *  *            !
                  *  *            !
                  *  *            !
                  *  *            !
                  *  *            !
                  *  *            !
              --->*  *<---T1      ! H                    SECTION PROPERTIES
                  *  *            !
                  *  *            !
                  *  *            !
           !      *  *            !
           !      *  *            !
           !      *  *            !
           V      *  *            !
          *********  *********    !
          *                  *    !
          *                  *    !
          ********************    V
           A
           !
           !
           ! T2

   ELEMENT      SECTION    H      B      T1     T2     A     IXX   ZEXX   ZPXX   RXX    IYY   ZEYY   ZPYY   RYY
       1       254.*146. 259.6  147.3   7.3   12.7   5.5   65.5  505.0  567.7  109.2   6.8   92.0  141.2   35.1

       2       254.*146. 259.6  147.3   7.3   12.7   5.5   65.5  505.0  567.7  109.2   6.8   92.0  141.2   35.1

       3       254.*146. 259.6  147.3   7.3   12.7   5.5   65.5  505.0  567.7  109.2   6.8   92.0  141.2   35.1

       4       254.*146. 259.6  147.3   7.3   12.7   5.5   65.5  505.0  567.7  109.2   6.8   92.0  141.2   35.1

       5       254.*146. 259.6  147.3   7.3   12.7   5.5   65.5  505.0  567.7  109.2   6.8   92.0  141.2   35.1

       6       254.*146. 259.6  147.3   7.3   12.7   5.5   65.5  505.0  567.7  109.2   6.8   92.0  141.2   35.1

       7       254.*146. 259.6  147.3   7.3   12.7   5.5   65.5  505.0  567.7  109.2   6.8   92.0  141.2   35.1

       8       254.*146. 259.6  147.3   7.3   12.7   5.5   65.5  505.0  567.7  109.2   6.8   92.0  141.2   35.1

       9       254.*146. 259.6  147.3   7.3   12.7   5.5   65.5  505.0  567.7  109.2   6.8   92.0  141.2   35.1

      10       254.*146. 259.6  147.3   7.3   12.7   5.5   65.5  505.0  567.7  109.2   6.8   92.0  141.2   35.1

      11       254.*146. 259.6  147.3   7.3   12.7   5.5   65.5  505.0  567.7  109.2   6.8   92.0  141.2   35.1

      12       254.*146. 259.6  147.3   7.3   12.7   5.5   65.5  505.0  567.7  109.2   6.8   92.0  141.2   35.1

      13       254.*146. 259.6  147.3   7.3   12.7   5.5   65.5  505.0  567.7  109.2   6.8   92.0  141.2   35.1
```

```
 14      254.*146.  259.6   147.3    7.3    12.7    5.5    65.5   505.0  567.7  109.2    6.8   92.0  141.2   35.1
 15      254.*146.  259.6   147.3    7.3    12.7    5.5    65.5   505.0  567.7  109.2    6.8   92.0  141.2   35.1

                                   *******************************
                                   COMMENCE DEFORMATION ANALYSIS
                                   *******************************
```

```
                              LOAD CASE   1 IS CRITICAL

                              SOLUTION AT WORKING LOADS
RESULTS AFTER   1  ITERATIONS
                                   NODE DISPLACEMENTS

                   NODE        DX            DY          ROTATION

                     1       .00000        .00000        .15135-01
                     2     -.48144-01    -.18222-03    -.13837-02
                     3     -.39574-01    -.26484-01    -.12591-01
                     4     -.28101-01    -.61572-01    -.87895-02
                     5     -.24034-01    -.74585-01     .33280-03
                     6     -.19400-01    -.59778-01     .92281-02
                     7     -.77928-02    -.24280-01     .12346-01
                     8      .58915-06    -.33649-03    -.59431-07
                     9      .00000        .00000       -.14710-06
                    10      .77941-02    -.24280-01    -.12346-01
                    11      .19402-01    -.59779-01    -.92281-02
                    12      .24035-01    -.74585-01    -.33274-03
                    13      .28102-01    -.61572-01     .87896-02
                    14      .39575-01    -.26483-01     .12591-01
                    15      .48145-01    -.18222-03     .13836-02
                    16      .00000        .00000       -.15135-01

                               INTERNAL NODE FORCES

        ELEMENT      AXIAL        SHEAR      BENDING(A)   YIELD MOM(A)  BENDING(B)   YIELD MOM(B)

           1      -.40376+02    .17322+02   -.22437-05    .14192+03   -.86611+02    .14192+03
           2      -.24604+02   -.25440+02    .86611+02    .14192+03   -.51985+01    .14192+03
           3      -.23506+02   -.12980+02    .51985+01    .14192+03    .36341+02    .14192+03
           4      -.14401+02   -.72565+00   -.36341+02    .14192+03    .38655+02    .14192+03
           5      -.14587+02    .13109+01   -.38655+02    .14192+03    .34474+02    .14192+03
           6      -.22698+02    .13563+02   -.34474+02    .14192+03   -.89307+01    .14192+03
           7      -.24797+02    .26022+02    .89307+01    .14192+03   -.92209+02    .14192+03
           8      -.74481+02   -.91646-04    .45894-03    .14192+03   -.71169-06    .14192+03
           9      -.24797+02   -.26022+02    .92209+02    .14192+03   -.89304+01    .14192+03
          10      -.22698+02   -.13563+02    .89304+01    .14192+03    .34475+02    .14192+03
          11      -.14587+02   -.13109+01   -.34475+02    .14192+03    .38655+02    .14192+03
          12      -.14401+02    .72571+00   -.38655+02    .14192+03    .36341+02    .14192+03
          13      -.23506+02    .12980+02   -.36341+02    .14192+03   -.51990+01    .14192+03
          14      -.24604+02    .25440+02    .51989+01    .14192+03   -.86612+02    .14192+03
          15      -.40376+02   -.17322+02    .86612+02    .14192+03   -.22439-05    .14192+03


                         COMMENCE DEFLECTION CHECKS
                         **************************

                              AT WORKING LOADS

  NODE   2
  DEFLECTION IS -.481441-01             PERMISSIBLE :  .900000-01
```

```
NODE  5
DEFLECTION IS -.746349-01           PERMISSIBLE :  .900000-01

NODE 12
DEFLECTION IS  .240355-01           PERMISSIBLE :  .900000-01

NODE 15
DEFLECTION IS -.182223-03           PERMISSIBLE :  .900000-01
```

```
                                                   SOLUTION AT FACTORED LOADS
     RESULTS AFTER  2  ITERATIONS
                                                         NODE DISPLACEMENTS
                          NODE           DX                  DY                 ROTATION

                            1          .00000              .00000              .29529-01
                            2        -.10253+00          -.31215-03            .33237-02
                            3        -.81350-01          -.65120-01          -.27151-01
                            4        -.57177-01          -.13895+00          -.18421-01
                            5        -.48257-01          -.16741+00          -.14099-04
                            6        -.39364-01          -.13904+00            .18393-01
                            7        -.15219-01          -.65291-01            .27123-01
                            8          .59352-02         -.57069-03          -.11870-02
                            9          .00000              .00000            -.11870-02
                           10          .27089-01         -.65292-01          -.27123-01
                           11          .51234-01         -.13904+00          -.18393-01
                           12          .60128-01         -.16741+00            .14025-04
                           13          .69048-01         -.13896+00            .18421-01
                           14          .93221-01         -.65121-01            .27151-01
                           15          .11440+00         -.31215-03          -.15607-01
                           16          .00000              .00000            -.31904-01

                                                     INTERNAL NODE FORCES

          ELEMENT        AXIAL            SHEAR          BENDING(A)        YIELD MOM(A)     BENDING(B)       YIELD MOM(B)

             1        -.68686+02        .28385+02      -.48876-05         .14192+03      -.14193+03        .14192+03
             2        -.44380+02       -.44075+02        .14193+03         .14192+03      -.87445+00        .14192+03
             3        -.37414+02       -.22894+02        .87446+00         .14192+03        .72391+02        .14192+03
             4        -.30428+02       -.20518+01      -.72391+02         .14192+03        .78934+02        .14192+03
             5        -.30427+02        .20520+01      -.78934+02         .14192+03        .72390+02        .14192+03
             6        -.37414+02        .22894+02      -.72390+02         .14192+03      -.87519+00        .14192+03
             7        -.44382+02        .44075+02        .87519+00         .14192+03      -.14192+03        .14192+03
             8        -.12557+03        .00000           .14305-05         .14192+03      -.16689-05        .14192+03
             9        -.44382+02       -.44075+02        .14193+03         .14192+03      -.87519+00        .14192+03
            10        -.37415+02       -.22894+02        .87522+00         .14192+03        .72390+02        .14192+03
            11        -.30427+02       -.20522+01      -.72390+02         .14192+03        .78935+02        .14192+03
            12        -.30427+02        .20518+01      -.78934+02         .14192+03        .72392+02        .14192+03
            13        -.37413+02        .22894+02      -.72392+02         .14192+03      -.87407+00        .14192+03
            14        -.44380+02        .44075+02        .87407+00         .14192+03      -.14192+03        .14192+03
            15        -.68685+02       -.28385+02        .14192+03         .14192+03        .00000           .14192+03

                                                     PLASTIC ROTATIONS
                         ELEMENT    NODE END              THETA(P)

                                       1                   .00000
                            1
                                       2                   .86286-03


                                       2                   .13043-01
                            2
                                       3                   .00000


                                       7                   .00000
```

```
                                          7
                                                     8               -.10870-01


                                                     8                .85039-02
                                          9
                                                    10                .00000


                                                    14                .00000
                                         14
                                                    15               -.25326-01


                                                    15               -.10772-01
                                         15
                                                    16                .00000





                                               COMMENCE DEFLECTION CHECKS
                                               **************************

                                                   AT ULTIMATE LOADS


      NODE   2
      DEFLECTION IS -.102532+00                PERMISSIBLE :  .200000+00


      NODE   5
      DEFLECTION IS -.167406+00                PERMISSIBLE :  .200000+00


      NODE  12
      DEFLECTION IS  .601275-01                PERMISSIBLE :  .200000+00


      NODE  15
      DEFLECTION IS -.312147-03                PERMISSIBLE :  .200000+00
```

```
    LOAD CASE   2   MUST NOT BE CHECKED


                                                            SOLUTION AT WORKING LOADS
  RESULTS AFTER   1   ITERATIONS
                                                                 NODE DISPLACEMENTS

                                             NODE            DX                 DY               ROTATION

                                               1          .00000            .00000            .58310-02
                                               2        -.97669-02        -.17406-03        -.58019-02
                                               3         .53917-03        -.31793-01        -.12130-01
                                               4         .10294-01        -.61557-01        -.61997-02
                                               5         .11872-01        -.66922-01         .23296-02
                                               6         .17485-01        -.48968-01         .93038-02
                                               7         .28210-01        -.16097-01         .10552-01
                                               8         .33293-01        -.33861-03        -.35792-02
                                               9         .00000            .00000           -.81982-02
                                              10         .42837-01        -.29603-01        -.12530-01
                                              11         .33575-01        -.62464-01        -.75639-02
                                              12         .36336-01        -.71362-01         .16348-02
                                              13         .51876-01        -.53620-01         .97516-02
                                              14         .73371-01        -.13511-01         .11461-01
                                              15         .79312-01        -.19026-03        -.29173-02
                                              16         .00000            .00000           -.22335-01

                                                                INTERNAL NODE FORCES

                  ELEMENT         AXIAL          SHEAR         BENDING(A)      YIELD MOM(A)    BENDING(B)      YIELD MOM(B)

                     1         -.34391+02     .12199+02     -.73360-06      .14192+03     -.60994+02      .14192+03
                     2         -.31553+02    -.21719+02      .60994+02      .14192+03      .91537+01      .14192+03
                     3         -.27460+02    -.94600+01     -.91536+01      .14192+03      .39428+02      .14192+03
                     4         -.24374+02     .27395+01     -.39428+02      .14192+03      .30692+02      .14192+03
                     5         -.24564+02     .12696+01     -.30672+02      .14192+03      .26644+02      .14192+03
                     6         -.24774+02     .13455+02     -.26644+02      .14192+03     -.16416+02      .14192+03
                     7         -.32872+02     .25914+02      .16416+02      .14192+03     -.99348+02      .14192+03
                     8         -.74507+02    -.46736+01      .24213+02      .14192+03      .00000         .14192+03
                     9         -.27157+02    -.24041+02      .75130+02      .14192+03      .18078+01      .14192+03
                    10         -.24053+02    -.11582+02     -.18078+01      .14192+03      .38873+02      .14192+03
                    11         -.14958+02     .66593+00     -.38873+02      .14192+03      .36750+02      .14192+03
                    12         -.14851+02     .21225+01     -.36750+02      .14192+03      .29981+02      .14192+03
                    13         -.23470+02     .14531+02     -.29981+02      .14192+03     -.15977+02      .14192+03
                    14         -.24395+02     .26621+02      .15977+02      .14192+03     -.10181+03      .14192+03
                    15         -.41866+02    -.20362+02      .10131+03      .14192+03      .00000         .14192+03


                                                        COMMENCE DEFLECTION CHECKS
                                                        **************************

                                                             AT WORKING LOADS

      NODE   2
      DEFLECTION IS -.97669-02                          PERM,SSIBLE :  .70000-01

      NODE   5
```

```
DEFLECTION IS -.669218-01          PERMISSIBLE :  .900000-01

NODE 12
DEFLECTION IS  .563357-01          PERMISSIBLE :  .900000-01

NODE 15
DEFLECTION IS -.190265-03          PERMISSIBLE :  .900000-01
```

```
                                          SOLUTION AT FACTORED LOADS
 RESULTS AFTER   1   ITERATIONS
                                                 NODE DISPLACEMENTS

                              NODE          DX              DY             ROTATION

                                1        .00000          .00000           .75803-02
                                2      -.12697-01      -.22629-03        -.75425-02
                                3       .70093-03      -.41339-01        -.15769-01
                                4       .13317-01      -.80029-01        -.80596-02
                                5       .15434-01      -.86999-01         .30285-02
                                6       .22730-01      -.63658-01         .12095-01
                                7       .36672-01      -.20926-01         .13718-01
                                8       .43280-01      -.44019-03        -.46530-02
                                9       .00000          .00000           -.10658-01
                               10       .55689-01      -.38490-01        -.16289-01
                               11       .69647-01      -.81203-01        -.98330-02
                               12       .73236-01      -.92770-01         .21252-02
                               13       .80464-01      -.69705-01         .12677-01
                               14       .95382-01      -.24065-01         .14900-01
                               15       .10311+00      -.24734-03        -.37925-02
                               16       .00000          .00000           -.29035-01

                                              INTERNAL NODE FORCES

         ELEMENT        AXIAL         SHEAR       BENDING(A)    YIELD MOM(A)   BENDING(B)    YIELD MOM(B)

            1        -.47791+02     .15858+02    -.95367-06     .14192+03     -.79292+02     .14192+03
            2        -.41026+02    -.28495+02     .79292+02     .14192+03      .11900+02     .14192+03
            3        -.35673+02    -.12270+02    -.11900+02     .14192+03      .51256+02     .14192+03
            4        -.30412+02     .35613+01    -.51256+02     .14192+03      .39900+02     .14192+03
            5        -.32063+02     .16505+01    -.39900+02     .14192+03      .34637+02     .14192+03
            6        -.37406+02     .17491+02    -.34637+02     .14192+03     -.21340+02     .14192+03
            7        -.47733+02     .33669+02     .21340+02     .14192+03     -.12915+03     .14192+03
            8        -.96860+02    -.62967+01     .31483+02     .14192+03      .00000        .14192+03
            9        -.35304+02    -.31254+02     .97670+02     .14192+03      .23502+01     .14192+03
           10        -.24975+02    -.15057+02    -.23501+01     .14192+03      .50535+02     .14192+03
           11        -.24657+02     .86571+00    -.50535+02     .14192+03      .47774+02     .14192+03
           12        -.24806+02     .27592+01    -.47774+02     .14192+03      .38976+02     .14192+03
           13        -.31162+02     .16669+02    -.38976+02     .14192+03     -.20771+02     .14192+03
           14        -.36487+02     .34867+02     .20771+02     .14192+03     -.13235+03     .14192+03
           15        -.54426+02    -.26471+02     .13235+03     .14192+03      .00000        .14192+03


                                         COMMENCE DEFLECTION CHECKS
                                         **************************

                                              AT ULTIMATE LOADS


     NODE   2
     DEFLECTION IS -.126969-01                   PERMISSIBLE :  .200000+00


     NODE   5
     DEFLECTION IS -.869983-01                   PERMISSIBLE :  .200000+00
```

```
NODE 12
DEFLECTION IS  .732364-01          PERMISSIBLE :  .200000+00

NODE 15
DEFLECTION IS -.247344-03          PERMISSIBLE :  .200000+00
```

```
                                        ***********************
                                        COMMENCE DESIGN CHECKS
                                        ***********************
                              ( IN ACCORDANCE WITH A.I.S.C. SPECIFICATIONS )


                                   LOCAL BUCKLING
ELEMENT  FLANGES(B/T RATIO) WEBS(H/T1 RATIO) SHEAR.WEB CRIPPLING EFF.YIELD MOMENT SLENDERNESS RATIO AXIAL STRNTH IN PLANE BUCKLING
         /----------------/ /--------------/ /-----------------/ /--------------/ /---------------/ /----------/ /---------------/

              SLENDERNESS RATIO UNACCEPTABLE FOR ELEMENT  1
   2  :         OK                OK               OK                 OK              N/A              N/A            OK

   3  :         OK                OK               OK                 OK              N/A              N/A            OK

   4  :         OK                OK               OK                 OK              N/A              N/A            OK

   5  :         OK                OK               OK                 OK              N/A              N/A            OK

   6  :         OK                OK               OK                 OK              N/A              N/A            OK

   7  :         OK                OK               OK                 OK              N/A              N/A            OK


              SLENDERNESS RATIO UNACCEPTABLE FOR ELEMENT  8
   9  :         OK                OK               OK                 OK              N/A              N/A            OK

  10  :         OK                OK               OK                 OK              N/A              N/A            OK

  11  :         OK                OK               OK                 OK              N/A              N/A            OK

  12  :         OK                OK               OK                 OK              N/A              N/A            OK

  13  :         OK                OK               OK                 OK              N/A              N/A            OK

  14  :         OK                OK               OK                 OK              N/A              N/A            OK


              SLENDERNESS RATIO UNACCEPTABLE FOR ELEMENT  15
```

```
                                             ************
                                             NEW SECTIONS
                                             ************

 ELEMENT     SECTION     H       B      T1     T2      A     IXX    ZEXX    ZPXX    RXX    IYY    ZEYY   ZPYY    RYY
    1      356.*171.  364.0   173.2    9.1   15.7    8.6   195.4  1073.0  1213.0  151.1   13.6   157.3  243.0   39.9
    2      254.*146.  259.6   147.3    7.3   12.7    5.5    65.5   505.0   567.7  109.2    6.8    92.0  141.2   35.1
    3      254.*146.  259.6   147.3    7.3   12.7    5.5    65.5   505.0   567.7  109.2    6.8    92.0  141.2   35.1
    4      254.*146.  259.6   147.3    7.3   12.7    5.5    65.5   505.0   567.7  109.2    6.8    92.0  141.2   35.1
    5      254.*146.  259.6   147.3    7.3   12.7    5.5    65.5   505.0   567.7  109.2    6.8    92.0  141.2   35.1
    6      254.*146.  259.6   147.3    7.3   12.7    5.5    65.5   505.0   567.7  109.2    6.8    92.0  141.2   35.1
    7      254.*146.  259.6   147.3    7.3   12.7    5.5    65.5   505.0   567.7  109.2    6.8    92.0  141.2   35.1
    8      356.*171.  364.0   173.2    9.1   15.7    8.6   195.4  1073.0  1213.0  151.1   13.6   157.3  243.0   39.9
    9      254.*146.  259.6   147.3    7.3   12.7    5.5    65.5   505.0   567.7  109.2    6.8    92.0  141.2   35.1
   10      254.*146.  259.6   147.3    7.3   12.7    5.5    65.5   505.0   567.7  109.2    6.8    92.0  141.2   35.1
   11      254.*146.  259.6   147.3    7.3   12.7    5.5    65.5   505.0   567.7  109.2    6.8    92.0  141.2   35.1
   12      254.*146.  259.6   147.3    7.3   12.7    5.5    65.5   505.0   567.7  109.2    6.8    92.0  141.2   35.1
   13      254.*146.  259.6   147.3    7.3   12.7    5.5    65.5   505.0   567.7  109.2    6.8    92.0  141.2   35.1
   14      254.*146.  259.6   147.3    7.3   12.7    5.5    65.5   505.0   567.7  109.2    6.8    92.0  141.2   35.1
   15      356.*171.  364.0   173.2    9.1   15.7    8.6   195.4  1073.0  1213.0  151.1   13.6   157.3  243.0   39.9



          HAVING OBTAINED NEW SECTIONS THE STRUCTURE MUST BE REANALYSED USING DEFORMATION THEORY


                                        SOLUTION AT WORKING LOADS
  RESULTS AFTER  1  ITERATIONS
                                              NODE DISPLACEMENTS

                         NODE        DX                  DY                  ROTATION

                          1        .00000              .00000              .97672-02
                          2      -.38494-01          -.12043-03            .35618-02
                          3      -.33659-01          -.15086-01           -.10025-01
                          4      -.23700-01          -.45593-01           -.82871-02
                          5      -.19263-01          -.59798-01           -.89705-03
                          6      -.16363-01          -.50446-01            .70862-02
                          7      -.68095-02          -.21178-01            .10610-01
                          8       .46338-06          -.21120-03           -.71305-07
```

```
                          9        .00000          .00000           -.10336-06
                         10        .33105-02      -.21173-01        -.10610-01
                         11        .16354-01      -.50445-01        -.73661-02
                         12        .19264-01      -.59793-01         .89713-03
                         13        .23701-01      -.45592-01         .82872-02
                         14        .33650-01      -.15086-01         .10025-01
                         15        .38495-01      -.12043-03        -.35619-02
                         16        .00000          .00000           -.97674-02

                                        INTERNAL NODE FORCES

     ELEMENT        AXIAL            SHEAR        BENDING(A)      YIELD MOM(A)    BENDING(B)     YIELD MOM(B)

        1         -.41205+02       .19401+02     -.44879-05       .30325+03     -.97004+02       .30325+03
        2         -.28925+02      -.25844+02      .97004+02       .14192+03     -.14298+02       .14192+03
        3         -.24827+02      -.13384+02      .14298+02       .14192+03      .28535+02       .14192+03
        4         -.20717+02      -.11553+01     -.28535+02       .14192+03      .32219+02       .14192+03
        5         -.20233+02      -.37392+00     -.32219+02       .14192+03      .33412+02       .14192+03
        6         -.24326+02       .11850+02     -.33412+02       .14192+03     -.45441+01       .14192+03
        7         -.28424+02       .24320+02      .45441+01       .14192+03     -.82373+02       .14192+03
        8         -.72263+02      -.10021-03      .50110-03       .30325+03     -.33589-07       .30325+03
        9         -.28424+02      -.24320+02      .82372+02       .14192+03     -.45437+01       .14192+03
       10         -.24326+02      -.11860+02      .45437+01       .14192+03      .33412+02       .14192+03
       11         -.20233+02       .37393+00     -.33412+02       .14192+03      .32219+02       .14192+03
       12         -.20717+02       .11554+01     -.32219+02       .14192+03      .28535+02       .14192+03
       13         -.24326+02       .13384+02     -.28535+02       .14192+03     -.14298+02       .14192+03
       14         -.28925+02       .25844+02      .14298+02       .14192+03     -.97004+02       .14192+03
       15         -.41205+02      -.19401+02      .97004+02       .30325+03      .44879-05       .30325+03



                               COMMENCE DEFLECTION CHECKS
                               **************************

                                    AT WORKING LOADS

   NODE   2
   DEFLECTION IS -.384737-01           PERMISSIBLE :  .900000-01

   NODE   5
   DEFLECTION IS -.597777-01           PERMISSIBLE :  .900000-01

   NODE  12
   DEFLECTION IS  .192644-01           PERMISSIBLE :  .900000-01

   NODE  15
   DEFLECTION IS -.128427-03           PERMISSIBLE :  .900000-01
```

```
                                         SOLUTION AT FACTORED LOADS
RESULTS AFTER  3  ITERATIONS
                                              NODE DISPLACEMENTS

                          NODE           DX               DY             ROTATION

                            1         .00000           .00000          .23279-01
                            2       -.10127+00       -.20074-03        .14200-01
                            3       -.80088-01       -.64994-01       -.27147-01
                            4       -.55919-01       -.13882+00       -.18416-01
                            5       -.47008-01       -.16725+00       -.90449-05
                            6       -.38106-01       -.13887+00        .18398-01
                            7       -.13955-01       -.65105-01        .27128-01
                            8        .72042-02       -.36701-03       -.14408-02
                            9        .00000           .00000          -.14408-02
                           10        .28364-01       -.65106-01       -.27129-01
                           11        .52514-01       -.13887+00       -.18398-01
                           12        .61413-01       -.16725+00        .90031-05
                           13        .70329-01       -.13882+00        .18416-01
                           14        .94497-01       -.64995-01        .27147-01
                           15        .11568+00       -.20074-03       -.17082-01
                           16        .00000           .00000          -.26161-01

                                           INTERNAL NODE FORCES

      ELEMENT        AXIAL          SHEAR        BENDING(A)    YIELD MOM(A)   BENDING(B)    YIELD MOM(B)

          1       -.68686+02      .28385+02     -.38147-05      .30325+03    -.14193+03      .30325+03
          2       -.44381+02     -.44075+02      .14193+03      .14192+03    -.97416+00      .14192+03
          3       -.37414+02     -.22894+02      .87418+00      .14192+03     .72392+02      .14192+03
          4       -.30427+02     -.20520+01     -.72392+02      .14192+03     .78935+02      .14192+03
          5       -.30427+02      .20522+01     -.78935+02      .14192+03     .72391+02      .14192+03
          6       -.37414+02      .22894+02     -.72391+02      .14192+03    -.87474+00      .14192+03
          7       -.44382+02      .44075+02      .87476+00      .14192+03    -.14192+03      .14192+03
          8       -.12557+03      .54836-05     -.19073-04      .30325+03    -.85831-05      .30325+03
          9       -.44382+02     -.44075+02      .14193+03      .14192+03    -.87455+00      .14192+03
         10       -.37414+02     -.22894+02      .87456+00      .14192+03     .72391+02      .14192+03
         11       -.30427+02     -.20523+01     -.72391+02      .14192+03     .78936+02      .14192+03
         12       -.30426+02      .20520+01     -.78936+02      .14192+03     .72392+02      .14192+03
         13       -.37412+02      .22894+02     -.72392+02      .14192+03    -.87390+00      .14192+03
         14       -.44379+02      .44075+02      .87391+00      .14192+03    -.14192+03      .14192+03
         15       -.68685+02     -.28385+02      .14193+03      .30325+03     .15259-04      .30325+03

                                              PLASTIC ROTATIONS
                          ELEMENT     NODE END              THETA(P)

                                          2                 .23915-01
                             2
                                          3                 .00000


                                          7                 .00000
                             7
                                          8                -.11137-01


                                          8                 .82558-02
```

```
                         9
                                    10                .00000

                                    14                .00000
                        14
                                    15               -.26797-01



                                 COMMENCE DEFLECTION CHECKS
                                 **************************

                                     AT ULTIMATE LOADS


    NODE   2
    DEFLECTION IS -.101265+00           PERMISSIBLE :  .200000+00


    NODE   5
    DEFLECTION IS -.167251+00           PERMISSIBLE :  .200000+00


    NODE  12
    DEFLECTION IS  .614126-01           PERMISSIBLE :  .200000+00


    NODE  15
    DEFLECTION IS -.200790-03           PERMISSIBLE :  .200000+00
```

```
                                                  ***********************
                                                  COMMENCE DESIGN CHECKS
                                                  ***********************
                                        ( IN ACCORDANCE WITH A.I.S.C. SPECIFICATIONS )


                                             LOCAL BUCKLING
 ELEMENT  FLANGES(B/T RATIO) WEBS(H/T1 RATIO) SHEAR,WEB CRIPPLING EFF.YIELD MOMENT SLENDERNESS RATIO AXIAL STRNTH IN PLANE BUCKLING
          /-----------------/ /--------------/ /-----------------/ /--------------/ /----------------/ /-----------/ /----------------/

    1  :        OK               OK                 OK                 OK              OK              OK              OK

    2  :        OK               OK                 OK                 OK              N/A             N/A             OK

    3  :        OK               OK                 OK                 OK              N/A             N/A             OK

    4  :        OK               OK                 OK                 OK              N/A             N/A             OK

    5  :        OK               OK                 OK                 OK              N/A             N/A             OK

    6  :        OK               OK                 OK                 OK              N/A             N/A             OK

    7  :        OK               OK                 OK                 OK              N/A             N/A             OK

    8  :        OK               OK                 OK                 OK              OK              OK              OK

    9  :        OK               OK                 OK                 OK              N/A             N/A             OK

   10  :        OK               OK                 OK                 OK              N/A             N/A             OK

   11  :        OK               OK                 OK                 OK              N/A             N/A             OK

   12  :        OK               OK                 OK                 OK              N/A             N/A             OK

   13  :        OK               OK                 OK                 OK              N/A             N/A             OK

   14  :        OK               OK                 OK                 OK              N/A             N/A             OK

   15  :        OK               OK                 OK                 OK              OK              OK              OK


                              ALL DESIGN CHECKS SUCCESSFULLY COMPLETED
```

APPENDIX A

Theoretical Basis of the Incremental Theory Analysis[A.1]

Consider a plane frame comprised of an elastic, perfectly plastic material. In this specific application the frame is subjected to monotorically increasing proportional loading, although the general theory is applicable to more complex loading programmes.

The frame is divided into uniform straight unloaded <u>elements</u> by an appropriate choice of <u>nodes</u>. External loads are applied at node points only, as are displacement constraints (support conditions).

The load vector at the k^{th} node is

$$\{P_k\} = \begin{bmatrix} V_k \\ H_k \\ M_k^{imp} \end{bmatrix} \qquad \text{A.1}$$

where V_k, H_k are respectively the applied loads in the global x and y directions, and M_k^{imp} is the applied moment, anticlockwise positive.

The displacement vector $\{u_k\}$ at the k^{th} node is

$$\{u_k\} = \begin{bmatrix} u_k \\ v_k \\ \theta_k \end{bmatrix} \qquad \text{A.2}$$

where u_k, v_k are respectively the displacement in the global x and y directions, and θ_k is the rotation, anticlockwise positive.

The deformations of an element connected at nodes j and k are given by the generalized element strain vector $\{\delta_{jk}\}$

$$\{\delta_{jk}\} = \begin{bmatrix} \delta_{jk} \\ \gamma_{jk} \\ \theta_{jk} \\ \theta_{kj} \end{bmatrix} \qquad \text{A.3}$$

where δ_{jk} is the extension in the direction $\vec{jk}$

γ_{jk} is the transverse displacement normal to $\vec{jk}$

θ_{jk} is the rotational strain at the j_{th} end of the element

θ_{kj} is the rotational strain at the k_{th} end of the element

The strain quantities are shown in Figure A.1

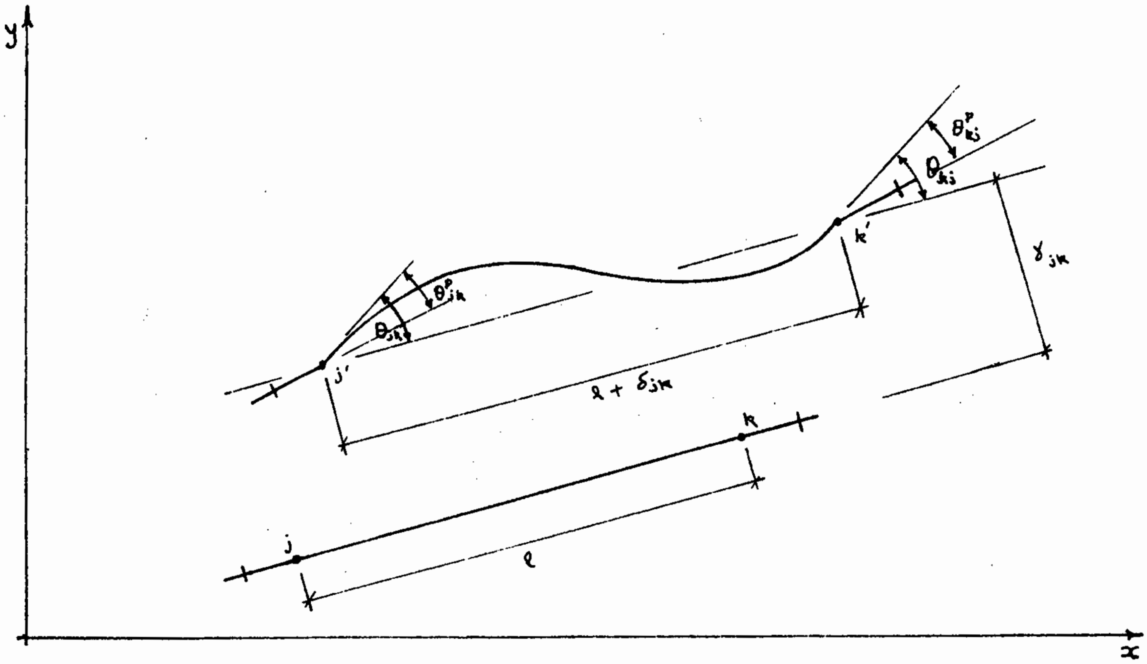


Figure A.1

The relation between $\{\delta_{jk}\}$ and $\{u_j\}$, $\{u_k\}$ is

$$\{\delta_{jk}\} = [B_{jk}] \begin{bmatrix} u_j \\ \cdots \\ u_k \end{bmatrix} \qquad A.4$$

where $[B_{jk}]$ is the element deformation matrix.

$$[B_{jk}] = \left[\begin{array}{ccc|ccc} -\text{Cos}\,\phi & -\text{Sin}\,\phi & 0 & \text{Cos}\,\phi & \text{Sin}\,\phi & 0 \\ +\text{Sin}\,\phi & -\text{Cos}\,\phi & 0 & -\text{Sin}\,\phi & \text{Cos}\,\phi & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right]$$

A.5

where ϕ is the angle between the global x axis and the direction $\vec{jk}$, anticlockwise positive.

The internal forces in element $_{jk}$ are

$$\{N_{jk}\} = \begin{bmatrix} N_{jk} \\ S_{jk} \\ M_j \\ M_k \end{bmatrix} \qquad A.6$$

These internal forces are shown in Figure A.2

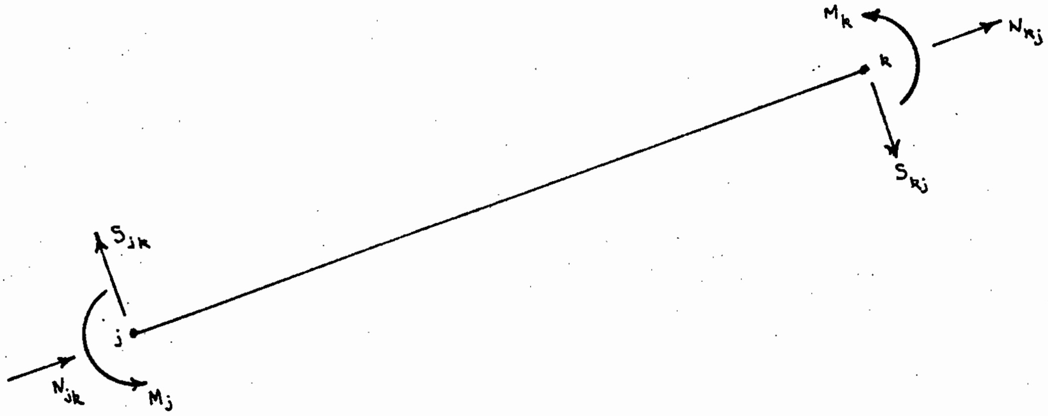


Figure A.2

The relation between bending moment M and curvature K at any point on an element is assumed to be elastic, perfectly plastic, with yielding under constant moment when $M = \pm M_p$. The relation between the elastic component of curvature K^e and bending moment M is

$$K^e = \frac{M}{EI}, \qquad \text{A.7}$$

where EI is the flexural rigidity. It is assumed that axial behaviour is entirely elastic, so that

$$\varepsilon = \frac{N}{AE}, \qquad \text{A.8}$$

where ε is the axial strain, AE is the axial rigidity, and the relation holds for all values of the axial force N. Shear deformation is neglected.

Because each element carries no transverse external load along its length, the bending moment is linear along the length. Peak values of moment occur only at the ends of the element adjacent to the nodes. Thus the behaviour of the element is elastic along its length: plastic deformation can occur only at the ends in the form of plastic hinges.

In order to express the stress-strain relations in matrix form, we define a <u>plastic strain</u> vector $\{\delta^p_{jk}\}$ for the element jk,

$$\{\delta^p_{jk}\} = \begin{bmatrix} 0 \\ 0 \\ \theta^p_{jk} \\ \theta_{kj} \end{bmatrix} , \qquad \text{A.9}$$

where δ^p_{jk}, δ^p_{kj} are respectively the plastic hinge rotations at the ends j and k of the element.

On integrating equations A.7 and A.8 along the element in the conventional manner, we may write

$$\{N_{jk}\} = [S_{jk}]\left(\{\delta_{jk}\} - \{\delta^p_{jk}\}\right) \qquad \text{A.10}$$

where $[S_{jk}]$ is the element stiffness matrix given by

$$[S_{jk}] = \begin{bmatrix} \frac{AE}{\ell} & 0 & 0 & 0 \\ 0 & \frac{12EI}{\ell^3} & -\frac{6EI}{\ell^2} & -\frac{6EI}{\ell^2} \\ 0 & -\frac{6EI}{\ell^2} & \frac{4EI}{\ell} & \frac{2EI}{\ell} \\ 0 & -\frac{6EI}{\ell^2} & \frac{2EI}{\ell} & \frac{4EI}{\ell} \end{bmatrix} \qquad \text{A.11}$$

At potential plastic hinges we have

$$\begin{aligned} \Delta\theta^{p} &\geq 0 \quad \text{if} \quad M = + M_{p} \\ \Delta\theta^{p} &= 0 \quad \text{if} \quad - M_{p} \leq M \leq M_{p} \\ \Delta\theta^{p} &\leq 0 \quad \text{if} \quad M = - M_{p} \end{aligned} \qquad \text{A.12}$$

The node load and displacement vectors are assembled into global load and displacement vectors as defined below;

$$\{P\} = \begin{bmatrix} \{P_i\} \\ \cdots\cdot \\ \\ \cdots\cdot \\ \{P_j\} \\ \cdots\cdot \\ \\ \cdots\cdot \\ \{P_n\} \end{bmatrix} , \qquad \text{A.13}$$

and

$$\{u\} = \begin{bmatrix} \{u_i\} \\ \cdots\cdot \\ \\ \cdots\cdot \\ \{u_j\} \\ \cdots\cdot \\ \\ \cdots\cdot \\ \{u_n\} \end{bmatrix} \qquad \text{A.14}$$

where n is the number of nodes.

Similarly, we define global strain, plastic strain and internal force vectors;

$$\{\delta\} = \begin{bmatrix} \{\delta_i\} \\ \\ \{\delta_j\} \\ \\ \{\delta_m\} \end{bmatrix}, \quad \{\delta^P\} = \begin{bmatrix} \{\delta^P_i\} \\ \\ \{\delta^P_j\} \\ \\ \{\delta^P_m\} \end{bmatrix}, \quad \{N\} = \begin{bmatrix} \{N_i\} \\ \\ \{N_j\} \\ \\ \{N_m\} \end{bmatrix}, \qquad A.15$$

where m is the number of elements.

Global deformation and system matrices [B] and [S] are also defined. These matrices are assembled from the element deformation and stiffness matrices.

$$\{\delta\} = [B]\,\{u\}. \qquad A.16$$

$$\{N\} = [S]\big(\{\delta\} - \{\delta^P\}\big). \qquad A.17$$

The form of equations (A.12) impose an <u>incremental</u> approach to the solution of the problem. In a generic increment, the vectors $\{P\}$, $\{u\}$, $\{\delta\}$, $\{\delta^P\}$ and N are known, and incremental loads $\{\Delta P\}$ are applied to the structure. We solve for the increments in displacements, strains, plastic strains and internal forces, denoted by $\{\delta u\}$, $\{\Delta\delta\}$, $\{\Delta\delta^P\}$ and $\{\Delta N\}$ respectively. The increments are added to the previous values, and the process is repeated. Initially the structure is unloaded and undeformed, and hence initial values of $\{P\}$, $\{u\}$, $\{\delta\}$ $\{\delta^P\}$ and $\{N\}$ are all zero. In practice the response will be piecewise linear, and thus the increment in load $\{\Delta P\}$ can be sealed up until a new hinge is formed.

A displacement method solution of the incremental problem (Martin A.1) is given in terms of $\{\Delta u\}$ and $\{\Delta\delta^P\}$ by the least value of the functional

$$\overline{U}^o_p = \tfrac{1}{2}\{\Delta\delta - \Delta\delta^p\}^T[S]\{\Delta\delta - \Delta\delta^p\} - \{\Delta u\}^T\{\Delta P\} \qquad \text{A.18}$$

subject to the constraints given in equation (A.12) applied to each potential hinge position. Using equation (A.16), $\overline{U}^o_p$ can be more conveniently written as

$$\overline{U}^o_p = \tfrac{1}{2}\big([B]\{\Delta u\} - \{\Delta\delta^p\}\big)^T[S]\big([B]\{\Delta u\} - \{\Delta\delta^p\}\big) - \{\Delta u\}^T\{\Delta P\}$$

$$= \tfrac{1}{2}\Big[\{\Delta u\}^T[B]^T[S][B]\{\Delta u\}\Big] - \tfrac{1}{2}\Big[\{\delta^p\}^T[S]\{\Delta\delta^p\}\Big] - \{\Delta u\}^T\{\Delta P\}$$

$$= \tfrac{1}{2}\begin{bmatrix}\Delta u\\ \cdots\\ \Delta\delta^p\end{bmatrix}^T [K] \begin{bmatrix}\Delta u\\ \cdots\\ \Delta\delta^p\end{bmatrix} - \{\Delta u\}^T\{\Delta P\}, \qquad \text{A.19}$$

where

$$[K] = \left[\begin{array}{c|c} [B]^T[S][B] & -\,[B]^T[S] \\ \hline -\,[S][B] & [S] \end{array}\right] \qquad \text{A.20}$$

The internal force increment may then be recovered from the incremental form of equation (A.17)

$$\{\Delta N\} = [S]\big([B]\{\Delta u\} - \{\Delta\delta^p\}\big) \qquad \text{A.21}$$

Algorithm for the minimisation of $\overline{U}_p^o$

The following algorithm for finding the least value of $\overline{U}_p^o$ has been given by Martin and Reddy [A.2]. It is based on the observation that if it is known a priori whether any hinge for which $M = \pm M_p$ will deform plastically or not, the constraints imposed on the minimisation fall away. The least value of the functional is given by

$$\frac{\partial \overline{U}_p^o}{\partial\{\Delta u\}} = \frac{\partial U_p^o}{\partial\{\Delta\delta^p\}} = 0 , \qquad A.22$$

which can be written as

$$[K] \begin{bmatrix} \{\Delta u\} \\ \cdots \\ \{\Delta\delta^p\} \end{bmatrix} = \begin{bmatrix} \Delta P \\ \cdots \\ 0 \end{bmatrix} . \qquad A.23$$

The algorithm proceeds by assuming that each hinge loads, solving equations (A.23), and checking the solution against the constraints. If the constraints are not satisfied, the guesses are modified. While convergence has not been proved, experience shows that convergence is rapid.

The steps in the algorithm are set out below. For proportional loading, the load vector ΔP may be assigned an arbitrarily chosen magnitude.

Step 0. For all hinge locations for which $M = \pm M_p$, assume $\Delta\theta^p \neq 0$.

Step 1. Eliminate rows and columns of [K] corresponding to all elements of $\{\Delta\delta^p\}$ for which $\Delta\theta^p$ is zero because $- M_p < M < M_p$ or because $\Delta\theta^p$ has been assumed zero.

Step 2. Invert [K] and solve equation (A.23) for $\{\Delta u\}$, $\{\Delta\delta^p\}$. Determine $\{\Delta N\}$ from equation (A.21).

Step 3. Check all hinge locations for which $M = \pm M_p$.

$$\text{If}\ \left.\begin{array}{l} \Delta\theta^p \geq 0 \text{ for } M = + M_p \\ \Delta\theta^p \leq 0 \text{ for } M = - M_p \end{array}\right] \quad \text{if it was assumed that } \Delta\theta^p \neq 0,$$

$$\left.\begin{array}{l} \Delta M \leq 0 \text{ for } M = + M_p \\ \Delta M \geq 0 \text{ for } M = - M_p \end{array}\right] \quad \text{if it was assumed that } \Delta\theta^p = 0,$$

go to step 4.

However, if it was assumed that $\Delta\theta^p \neq 0$ at any hinge and

$$\left.\begin{array}{ll} \Delta\theta^p < 0, & M = + M_p \\ \Delta\theta^p > 0, & M = - M_p \end{array}\right] \quad \text{assume } \Delta\theta^p = 0 \text{ at that hinge,}$$

and if it was assumed that $\Delta\theta^p = 0$ at any hinge and

$$\left.\begin{array}{ll} \Delta M > 0, & M = + M_p \\ \Delta M < 0, & M = - M_p \end{array}\right] \quad \text{assume that } \Delta\theta^p \neq 0.$$

Return to Step 1.

Step 4. Choose the magnitude of ΔP by the condition that, among the hinge locations for which $|M| < M_p$, one location will have $|M| = M_p$ while $|M| \leq M_p$ for all others.

Step 5. Update $\{P\}$, $\{u\}$, $\{\delta^p\}$, $\{N\}$ by adding increments to starting values. Return to Step 0.

The iterative procedure is terminated when flow occurs, indicated by a very large increase in $\{u\}$, or by instability of equation (A.23).

Bibliography

A.1 J.B. Martin — Plasticity: Fundamentals and General Results. M.I.T. Press, 1975

A.2 J.B. Martin and B.D. Reddy — 1976 — 'A programming approach to the role problem in elastic-plastic solids', J. Eng. Mech. Div., ASCE, to be published. Presented at the ASCE Engineering Mechanics Speciality Conference, Waterloo, 1976.

APPENDIX B

Theoretical Basis of the Deformation Theory Analysis

The approach followed in this analysis is that of Baggett, Martin and Reddy [B.1].

The essential difference from incremental analysis is that the bilinear elastic, perfectly plastic moment-curvature relation (Figure B.1) is assumed to be reversible i.e. unloading occurs along the same curve. It follows then that deformation theory analysis will be correct only when unloading does not take place in any section at which plastic deformation takes place.

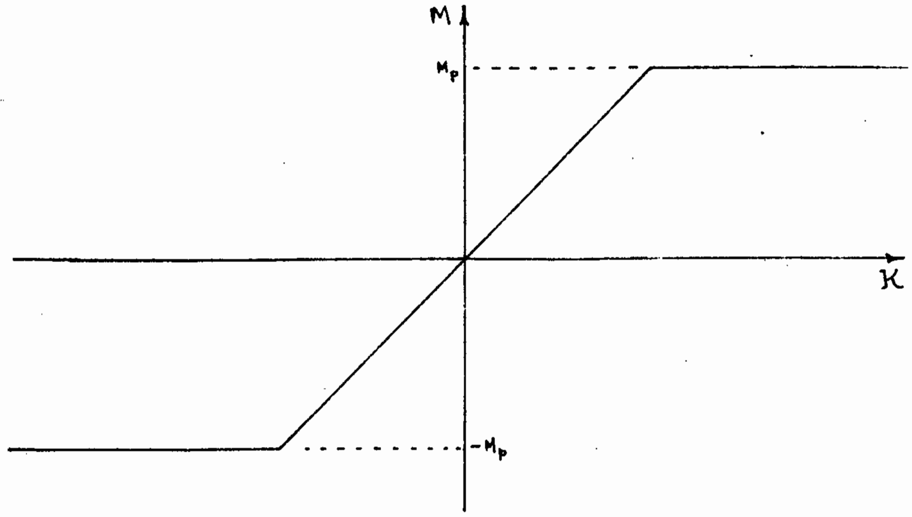


Figure B.1

In the context of the plane frame problem under discussion, the equations given in Appendix A hold except that equations A.12 are replaced by

$$\begin{aligned} &\theta^p \geq 0 \quad \text{if} \quad M = + M_p, \\ &\theta^p = 0 \quad \text{if} \ - M_p < M < M_p, \\ &\theta^p \leq 0 \quad \text{if} \quad M = - M_p. \end{aligned} \qquad \text{B.1}$$

It is no longer necessary to consider increments of load; we may impose the final loads {P} directly. In doing so we note that deformation theory analysis is not suitable for determining the limit loads (i.e. loads causing flow in the structure).

The deformation theory solution is given by the least value of the functional

$$U_p^o = \sum_{\text{hinges}} M_p|\theta^p| + \tfrac{1}{2}\begin{bmatrix}\{u\}\\ \cdots\\ \{\delta^p\}\end{bmatrix}[K]\left(\{u\}:\{\delta^p\}\right) - \{u\}^T\{P\}, \qquad \text{B.2}$$

subject to the constraints of equations B.1. The internal forces are given directly by

$$\{N\} = [S_e]\left([B]\{u\} - \{\delta^p\}\right). \qquad \text{B.3}$$

The first term in equations B.2 must be placed in matrix form. We write, at any one hinge,

$$M_p|\theta^p| = M_p.\theta^p.\lambda \qquad \text{B.4}$$

where

$$\begin{aligned} \lambda &= +1 \quad \text{if} \quad M = + M_p \\ \lambda &= 0 \quad \text{if} \ - M_p < M < M_p \\ \lambda &= -1 \quad \text{if} \quad M = - M_p \end{aligned} \qquad \text{B.5}$$

Thus for each element kℓ, we define

$$\{\hat{M}\}_{k\ell} = \begin{bmatrix} 0 \\ 0 \\ \lambda_{k\ell}M_p \\ \lambda_{\ell k}M_p \end{bmatrix} \qquad \text{B.6}$$

For the entire structure,

$$\{\hat{M}\} = \begin{bmatrix} \{\hat{M}_i\} \\ \cdots\cdot \\ \cdots\cdot \\ \{\hat{M}_j\} \\ \cdots\cdot \\ \cdots\cdot \\ \{\hat{M}_m\} \end{bmatrix} \qquad \text{B.7}$$

where m is the number of elements of the structure.

Equation B.2 may now be expressed in matrix form as:

$$\overline{U}^{o}_{p} = \{\delta^{p}\}^{T}\{\hat{M}\} + \tfrac{1}{2}\begin{bmatrix} \{u\} \\ \cdots\cdot \\ \{\delta p\} \end{bmatrix} [K] \begin{bmatrix} \{u\} \\ \cdots\cdot \\ \{\delta^{p}\} \end{bmatrix} - \{u\}^{T}\{P\} \qquad \text{B.8}$$

where [K] is defined by equation A.20.

The solution of a particular elastic-perfectly plastic problem is given by the least value of $\overline{U}^{o}_{p}$, with variables {u} and $\{\delta^{p}\}$, subject to the constrainsts of equation B.5.

The Algorithm for the Minimization of $\overline{U}_p^o$

Since $\{\delta^p\}$ is discontinuous when $|\delta^p| > 0$, simply setting

$$\frac{\partial \overline{U}_p^o}{\partial\{\delta^p\}} = 0$$ and solving the resulting set of simultaneous equations is inadmissible.

This problem may be circumvented by assuming some value for λ (Equations B.5), taking the partial derivatives (now continuous because the signum function has been eliminated) and solving the resulting system of simultaneous equations for $\{u\}$ and $\{\delta^p\}$. A check on the correctness of the assumed λ's is then required.

Thus, regarding $\{\hat{M}\}$ as known, the conditions for $\overline{U}_p^o$ to be a minimum is:

$$[K]\begin{bmatrix}\{u\}\\ \cdots\cdots\\ \{\delta^p\}\end{bmatrix} = \begin{bmatrix}\{P\}\\ \cdots\cdots\\ -\{\hat{M}\}\end{bmatrix} \qquad \text{B.9}$$

In solving equation B.9 for $\{u\}$ and $\{\delta^p\}$, all terms associated with elements of $\{\ddot{u}\}$ which are constrained to be zero (boundary conditions) or terms associated with elements of $\{\hat{M}\}$ which are assumed to be zero are ignored. This corresponds to eliminating a row and column of $[K]$ for each zero variable, and is implied wherever $[K]$ is inverted in the iterative process set out below:

1. Set all plastic hinge rotations to zero and solve equation B.9. This yields an elastic solution.
2. Consider each element and: if

$$M = M_p \text{ and } \theta^p > 0$$

or $$-M_p < M < M_p \text{ and } \theta^p = 0 \qquad \text{B.10}$$

or $$M = -M_p \text{ and } \theta^p < 0$$

at every member end throughout the structure, the solution has been found and the iteration procedure is stopped.

If however, at any member end:

$$M \geq M_p \quad \text{and} \quad \theta^p = 0; \quad \text{assume } \theta^p = 0 \text{ and set } \lambda = 1$$

B.11

$$M \leq -M_p \quad \text{and} \quad \theta^p = 0; \quad \text{assume } \theta^p = 0 \text{ and set } \lambda = -1$$

Alternatively, if at any member end:

$$\lambda = +1 \quad \text{and } \theta^p < 0; \quad \text{set } \theta^p = 0$$

B.12

$$\lambda = -1 \quad \text{and } \theta^p > 0; \quad \text{set } \theta^p = 0$$

3. Solve equation B.9 with the assigned values of α included, and determine $\{u\}$, $\{\delta^p\}$. Obtain the member end moments using equations B.3, and return to Step 2.

Baggett, Martin and Reddy[B.1] have been unable to give a rigorous proof for the convergence of this algorithm. Numerous examples worked by the writer confirm however that the algorithm converges rapidly to a solution.

Bibliography

B.1	J. Baggett, J.B. Martin and B.D. Reddy	1977	'Elastic-plastic deformations in plane frames', The Civil Engineer in South Africa, 19, 89-93.

APPENDIX C

ADPF - Program Listing

```
PLASLM*ADPF(1),MAIN
      COMPILER (XM=3)
      COMMON /EXT/RK(157,98)
C     ************************
C     CONTROLLING SUBROUTINE
C     ************************
      COMMON/BLK1/NROWP,NCOLP,MP(23,2),NF,NDF,U(73),P(73),
     #NE,IPFLTS(23,4,2),PSTRN(23,4),STRN(23,4),STRS(23,4),
     #NBWA,ISIG(23,4),NELTS,LP,NB,M,ISTOP,MAX,
     #NBW,NM,E,EP,AE,EI,YSTRS,NBEAM(23,2),COORDX(23),
     #COORDY(23),ELLEN(23),SELT(23),CELT(23),NDFA,NDFS,
     #NBWS,NDFBWA,IBC(73),N1,N2,IIBC(3),PP(10,73),RLAMDA(10),
     #PPP(3),WL(3),WLOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
     #B(23,4,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALPHA(23,2),
     #DSTRN(23,4),DPSTRN(23,4),DSTRS(23,4),NELT(88,4),DU(73),
     #DP(73),RMIN,IDUM,P1(73),MS,NBS,EA(23),RE1(23),IPHASE,
     #INDEX,IDUM1,LDCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
     #IEND(23),NLC,NSEC
      COMMON/BLK9/IBP,ISHORT,ITAB
      IPHASE=0
      NF=3
C     *************************
C     READ IN AND DISPLAY DATA
C     *************************
      CALL DATA
C     ****************************************
C     IF DEFORMATION ANALYSIS ALONE IS REQUIRED
C     INCREMENTAL ANALYSIS IS OMITTED
C     ****************************************
      IF(IBP.EQ.1) CALL CONTRL
C     **********************************
C     SET UP ELEMENT STIFFNESS MATRICES
C     **********************************
      CALL ELTMAT
      LP=1
      NELTS=0
      NB=NBWA
      M=NDF
C     *******************************
C     ADD IN PROPORTIONAL LOAD VECTOR
C     *******************************
      DO 3 I=1,NDF
    3 RK(I,NB)=PP(LP,I)
C     *****************************************
C     SOLVE THE EXTENDED SYSTEM STIFFNESS MATRIX
C     FOR DISPLACEMENTS AND PLASTIC ROTATIONS
C     *****************************************
    4 CALL SOLVE
C     ************************************
C     DETERMINE STRAINS AND INTERNAL FORCES
C     ************************************
    1 CALL STRSTR
      IF(NELTS.EQ.0) GO TO 4
      CALL ITER
      GO TO 1
      END
```

```
[illegible],ADPF(1),DATA
              COMPILER ([illegible])
              SUBROUTINE DATA
              COMMON  /EXT/RK(157,93)
        C     *************************
        C     READS AND DISPLAYS DATA
        C     *************************
              COMMON /BLK1/NMJP,NCOLP,NP(23,2),NF,NDF,U(73),P(73),
             #NE,IPELTS(23,6,2),PSTRM(23,4),STRN(23,4),STPS(23,4),
             #NB,A,I,IG(23,6),NELTS,LP,NB,H,ISTOP,MAX,
             #IB,D,E,SP,AL,EI,YSTRS,NDFSM(23,2),COORDX(23),
             #COORDY(23),ELCEN(23),SELT(23),CELT(23),NDFA,NDFS,
             #NOS,NDFBA,IJC(73),N1,N2,IJOC(3),PP(10,73),RLAMDA(10),
             #PPP(3),JL(3),LOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
             #B(23,6,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALPHA(23,2),
             #DSTRN(23,4),DSTRM(23,4),DSTRS(23,4),NELT(83,4),DU(73),
             #DP(73),KTIN,I,NP,PI(73),NS,NNS,EA(23),REI(23),IPHASE,
             #INDEX,IOUT1,LOCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
             #IEND(23),NLC,NSEC
              COMMON /BLK2/CSTRS(10,23,4),EFFLEN(23),ICOL(10),
             #CHI(23),CH(23),PE(23),BUKCHK(23,2),BM(23),RMM(23),
             #OMP(23),RETURN(10,23)
              COMMON /BLK5/SSP(23,20)
              COMMON /BLK8/DEF(10,73),DEFCHU(20,3),NCHU,NCH,DEFCH(20,3)
              COMMON /BLK9/ISP,ISHORT,ITAB
              DIMENSION NPD(23),SSPD(20),IC(10)
              REAL IP,NPD
              NF=3
              MAX=0
              NBB=0
          100 FORMAT()
        C     ******************
        C     MAIN HEADER TITLE
        C     ******************
              PRINT 931
        931   FORMAT(1H1,////,1H ,46X,33('*'),/,1H ,46X,'*',31X,
             #'*',/,1H ,46X,'*',1X,'AUTOMATIC DESIGN AND ANALYSIS',
             #1X,'*',/,1H ,46X,'*',9X,'OF PLANE FRAMES',7X,'*',/,
             #1H ,46X,'*',31X,'*',/,1H ,46X,'*',11X,'P.D.GRIFFIN',
             #9X,'*',/,1H ,46X,'*',3X,'DEPT. OF CIVIL ENGINEERING',
             #2X,'*',/,1H ,46X,'*',4X,'UNIVERSITY OF CAPE TOWN',4X,
             #'*',/,1H ,46X,'*',31X,'*',/,1H ,46X,'*',31X,'*',/,
             #1H ,46X,'*',8X,'UNIVAC 1106 EXEC8',5X,'*',/,1H ,46X,'*',31X,
             #'*',/,1H ,
             #46X,33('*'),///,1H ,46X,33('*'),/,1H ,46X,'*',1X,
             #'JOB DESCRIPTION :',13X,'*',/,3(1H ,46X,'*',31X,
             #'*',/),1H ,46X,'*',1X,'DATE :........  USER :......... *',
             #/,1H ,46X,33('*'),////)
              PRINT 101
        101   FORMAT(1H1,45X,' UNITS : KILONEWTONS AND METRES',/,47X,30('*')
             #,////)
        C     **********
        C     DATA INPUT
        C     **********
              READ 100,NMJP,NCOLP,NSEC,ISP,ISHORT,ITAB
              READ 100,NE,NB,NLC,E,AL,EI,YSTRS
              READ 100,(RLAMDA(I),I=1,NLC)
              READ 100,((EFFLEN(IE),CHI(IE)),IE=1,NF)
              READ 100,NC,(IC(I),I=1,NC)
              READ 100,(((NBEAM(I,J),J=1,2),I=1,NE))
```

```
      READ 100,(MPD(IE),IE=1,NE)
      IF(IBP.NE.1) GO TO 102
      DO 103 IE=1,NE
  103 MPD(IE)=0
C     *****************************************
C     CALCULATES LENTHS,COS,SIN, FOR EACH MEMBER
C     *****************************************
  102 READ 100,((COORDX(I),COORDY(I)),I=1,NN)
      DO 112 I=1,NC
      IA=IC(I)
  112 ICOL(IA)=IC(I)
      DO 111 IK=1,NE
      DO 111 I=1,2
      MP(IK,I)=MPD(IK)
      IP=NBEAM(IK,1)
      IQ=NBEAM(IK,2)
      ELLEN(IK)=SQRT((COORDX(IP)-COORDX(IQ))**2.+
     #(COORDY(IP)-COORDY(IQ))**2.)
      CELT(IK)=(COORDX(IQ)-COORDX(IP))/ELLEN(IK)
      SELT(IK)=(COORDY(IQ)-COORDY(IP))/ELLEN(IK)
  111 CONTINUE
      DO 12 I=1,NE
      J1=ABS(NBEAM(I,1)-NBEAM(I,2))
      IF(J1.GT.MAX)MAX=J1
   12 IF(NBW.LT.MAX) NBW=MAX
      NBW=(NBW+1)*NF
      NDF=NJ*NF
      J1=NROWP-NDF
      J2=NCOLP-NBW-1
      MPE=MIN0(J1,J2)
      NDFA=NDF+1
      NDFS=NDF-1
      NBWA=NBW+1
      NBWS=NBWA-1
      NDFBWA=NDF-NBWS+1
C     **************
C     DISPLAYS DATA
C     **************
      PRINT 202,NE,NN
  202 FORMAT(//,' NUMBER OF ELEMENTS =',I5,/,
     $' NUMBER OF NODES    =',I5)
      EE=E/1.E3
      PRINT 203,EE
  203 FORMAT(/,' ELASTIC MODULUS  =',E11.6,/)
      PRINT 204
  204 FORMAT(//,6X,' COORDINATES OF NODES',//,' NODE',7X,'X',12X,'Y')
      DO 17 I=1,NN
   17 PRINT 205,I,COORDX(I),COORDY(I)
  205 FORMAT(1H ,I3,1X,2(2X,E11.6))
      PRINT 206
  206 FORMAT(//,' ELEMENT',4X,'NODES',8X,'LENGTH',
     #,/,10X,'/',7('-'),'/',/)
      PRINT 207,((I,(NBEAM(I,J),J=1,2),ELLEN(I)),I=1,NE)
  207 FORMAT(1H ,I4,1X,2I6,4X,E11.6)
      PRINT 208
  208 FORMAT(//,' BOUNDARY CONDITIONS : 0=FIXITY , 1=FREEDOM',
     #//,' NODE',3X,'X',3X,'Y',3X,'ROTATION')
      DO 67 I=1,NDF
   67 IBC(I)=1
    4 READ 100,N1,N2
```

```
           IF(N1.LE.0) GO TO 2
           READ 100,(IIBC(J),J=1,NF)
           DO 7 I=N1,N2
    7      PRINT 209,I,(IIBC(J),J=1,NF)
       209 FORMAT(1H ,I3,4X,I1,3X,I1,6X,I1)
           JJ=NF*(N1-1)
           II=N2-N1+1
           DO 3 I=1,II
           DO 3 J=1,NF
           JJ=JJ+1
    3      IBC(JJ)=IIBC(J)
           GO TO 4
    2      PRINT 210
       210 FORMAT(//,' LOADING PROGRAMME:')
           DO 21 I=1,NLC
           DO 21 J=1,NDF
    21     PP(I,J)=0.
           IK=0
    9      READ 100,IJ
           IK=IK+1
           IF(IJ.LE.-1) GO TO 5
           PRINT 220,IJ
    220    FORMAT(//' LOADING',I4)
           PRINT 218
    218    FORMAT(//,15X,'NODE',7X,'PX',11X,'PY',7X,'MOMENT')
    19     CONTINUE
           READ 100,N1,N2
           IF(N1.LE.0) GO TO 9
           READ 100,(PPP(J),J=1,NF)
           DO 11 I=N1,N2
    11     PRINT 219,I,(PPP(J),J=1,NF)
    219    FORMAT(15X,I3,1X,3(2X,E11.6))
    20     CONTINUE
           JJ=NF*(N1-1)
           I1=(N2-N1+1)
           DO 16 I=1,II
           DO 16 J=1,NF
           JJ=JJ+1
           PP(IJ,JJ)=PP(IJ,JJ)+PPP(J)*RLAMDA(IJ)/100.
    16     WLOAD(IJ,JJ)=WLOAD(IJ,JJ)+PP(IJ,JJ)*100.
           GO TO 19
    5      CONTINUE
           PRINT 303
    303    FORMAT(1H ,//,10X,' SELF WEIGHT IS AUTOMATICALLY'
         #' INCLUDED INTO THE LOAD VECTORS BEFORE DEFORMATION '
         #'ANALYSIS COMMENCES')
           IF(IBP.NE.1.AND.ISHORT.EQ.1) PRINT 104
    104    FORMAT(1H ,////,40X,31('*'),/,40X,'RESULTS OF'
         #' INCREMENTAL ANALYSIS',/,40X,31('*'),//)
           NCH=0
    301    READ 100,I
           IF(I.EQ.-1) GO TO 302
           NCH=NCH+1
           READ(5,100),I,J,DEFCH(NCH,3)
           DEFCH(NCH,1)=I
           DEFCH(NCH,2)=J
           GO TO 301
    302    CONTINUE
           NCHU=0
    411    READ 100,I
```

```
      IF(I.EQ.-1) GO TO 412
      NCHU=NCHU+1
      READ(0,100),I,J,DEFCHU(NCHU,3)
      DEFCHU(NCHU,1)=I
      DEFCHU(NCHU,2)=J
      GO TO 411
  412 CONTINUE
      IF(IBP,NE.1) GO TO 993
      IDUM=100
  199 READ 100,N1,N2
      IF(N1.LT.0) GO TO 992
      READ 100,((SSPD(I),I=1,20))
      DO 243 I=N1,N2
      DO 243 J=1,20
  243 SSP(I,J)=SSPD(J)
      GO TO 199
  992 CALL SORT
  993 IDUM=0
      IF(ISHORT.NE.1.AND.IBP.NE.1) PRINT 212
  212 FORMAT(1H ,//,' INCREMENTAL ANALYSIS OF PLANE FRAME USING BEAM
     #' ELEMENT',/)
      IF(IBP.NE.1.AND.ISHORT.NE.1) PRINT 300
  300 FORMAT(1H ,///,51X,'COMMENCE LOAD CASE 1',//)
      RETURN
      END
```

```
PLASLM*ADPF(1).SOLVE
              COMPILER (XM=3)
              SUBROUTINE SOLVE
              COMMON /EXT/RK(157,98)
C             ****************************************
C             SOLUTION ROUTINE  :  GAUSS-JORDAN REDUCTION
C             ****************************************
              COMMON/BLK1/NROWP,NCOLP,MP(23,2),NF,NDF,U(73),P(73),
             #NE,IPELTS(23,4,2),PSTRN(23,4),STRN(23,4),STRS(23,4),
             #NBWA,ISIG(23,4),NELTS,LP,NB,M,ISTOP,MAX,
             #NBW,NJ,E,EP,AE,EI,YSTRS,NBEAM(23,2),COORDX(23),
             #COORDY(23),ELLEN(23),SELT(23),CELT(23),NDFA,NDFS,
             #NBWS,NDFBWA,IBC(73),N1,N2,IIBC(3),PP(10,73),RLAMDA(10),
             #PPP(3),NL(3),WLOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
             #B(23,4,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALPHA(23,2),
             #DSTRN(23,4),DPSTRN(23,4),DSTRS(23,4),NELT(88,4),DU(73),
             #DP(73),RMIN,IDUM,P1(73),MS,NBS,EA(23),REI(23),IPHASE,
             #INDEX,IDUM1,LDCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
             #IEND(23),NLC,NSEC
              REWIND 13
              READ(13) ((RK(I,J),J=1,NBW),I=1,NDF)
              ISTOP=ISTOP+1
              MS=M-1
              NBS=NB-1
              DO 22 I=1,NDF
              IF(IBC(I).EQ.1) GO TO 22
              DO 23 J=NBWA,NB
           23 RK(I,J)=0.
           22 CONTINUE
              DET=1.
              DO 1 IR=1,NDF
              IF(IR.EQ.NDF) GO TO 10
              IB=IR+1
              IBB=IR+NBWS
              IF(IBB.GT.NDF) IBB=NDF
              NCOLL=NBWS
              IC=2
              DO 8 I=IB,IBB
              ICC=IC
              IF(RK(IR,IC).EQ.0) GO TO 20
              FACT=RK(IR,IC)/RK(IR,1)
              IF(IR.GE.NDFBWA) NCOLL=NDFA-I
              DO 9 J=1,NCOLL
              RK(I,J)=RK(I,J)-FACT*RK(IR,ICC)
            9 ICC=ICC+1
              DO 16 J=NBWA,NB
           16 RK(I,J)=RK(I,J)-FACT*RK(IR,J)
           20 NCOLL=NCOLL-1
            8 IC=IC+1
           10 DET=DET*(RK(IR,1))
              IF(NELTS.EQ.0) GO TO 1
              NCOLF=NBWA
              DO 15 I=NDFA,M
              IF(RK(IR,NCOLF).EQ.0) GO TO 15
              FACT=RK(IR,NCOLF)/RK(IR,1)
              DO 17 J=NCOLF,NB
           17 RK(I,J)=RK(I,J)-FACT*RK(IR,J)
           15 NCOLF=NCOLF+1
            1 CONTINUE
              IF(NELTS.EQ.0) GO TO 21
```

```
      IF(NELTS.EQ.1) GO TO 11
      NCOLF=NBWA
      DO 2 IR=NDFA,MS
      IRA=IR+1
      IC=NCOLF+1
      DO 3 I=IRA,M
      IF(RK(IR,IC).EQ.0) GO TO 3
      FACT=RK(IR,IC)/RK(IR,NCOLF)
      DO 4 J=IC,NB
    4 RK(I,J)=RK(I,J)-FACT*RK(IR,J)
    3 IC=IC+1
      DET=DET*(RK(IR,NCOLF))
    2 NCOLF=NCOLF+1
   11 CONTINUE
      RK(M,NB)=RK(M,NB)/RK(M,NBS)
      DET=DET*(RK(M,NBS))
      IF(NELTS.EQ.1) GO TO 7
      NCOLF=NBS
      DO 5 IR=MS,NDFA,-1
      II=IR+1
      DO 6 J=NCOLF,NBS
      RK(IR,NB)=RK(IR,NB)-RK(IR,J)*RK(II,NB)
    6 II=II+1
      NCOLF=NCOLF-1
    5 RK(IR,NB)=RK(IR,NB)/RK(IR,NCOLF)
    7 II=NDFA
      DO 12 J=NBWA,NBS
      DO 13 I=1,NDF
   13 RK(I,NB)=RK(I,NB)-RK(I,J)*RK(II,NB)
   12 II=II+1
   21 RK(NDF,NB)=RK(NDF,NB)/RK(NDF,1)
      DO 14 IR=NDFS,1,-1
      II=IR+1
      NCOLL=NBW
      IF(IR.GE.NDFBWA) NCOLL=NDFA-IR
      DO 19 J=2,NCOLL
      RK(IR,NB)=RK(IR,NB)-RK(IR,J)*RK(II,NB)
   19 II=II+1
   14 RK(IR,NB)=RK(IR,NB)/RK(IR,1)
      OUT(ISTOP)=DET
      RETURN
      END
```

```
PLASLM*ADPF(1).ELTMAT
      COMPILER (XM=3)
      SUBROUTINE ELTMAT
      COMMON /EXT/RK(157,98)
C     ***********************************************
C     SETS UP DEFORMATION AND STIFFNESS MATRICES FOR EACH ELEMENT
C     ***********************************************
      COMMON/BLK1/NROWP,NCOLP,NP(23,2),NF,NDF,U(73),P(73),
     #NE,IPELTS(23,4,2),PSTRN(23,4),STRN(23,4),STRS(23,4),
     #NBWA,ISIG(23,4),NELTS,LP,NB,M,ISTOP,MAX,
     #NBW,NW,E,EP,AE,EI,YSTRS,NBEAM(23,2),COORDX(23),
     #COORDY(23),ELLEN(23),SELT(23),CELT(23),NDFA,NDFS,
     #NBWS,NDFBWA,IBC(73),N1,N2,IIBC(3),PP(10,73),RLAMDA(10),
     #PPP(3),WL(3),WLOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
     #B(23,4,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALPHA(23,2),
     #DSTRN(23,4),DPSTRN(23,4),DSTRS(23,4),NELT(88,4),DU(73),
     #DP(73),RMIN,IDUM,P1(73),MS,NBS,EA(23),REI(23),IPHASE,
     #INDEX,IDUM1,LOCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
     #IEND(23),NLC,NSEC
      DO 2 IE=1,NE
      DO 1 I1=1,4
      DO 3 I2=1,4
    3 S(IE,I1,I2)=0.
C     **********
C     ELEMENT B MATRICES
      DO 1 I3=1,6
    1 B(IE,I1,I3)=0.
      B(IE,1,1)=-CELT(IE)
      B(IE,1,2)=-SELT(IE)
      B(IE,2,1)=SELT(IE)
      B(IE,2,2)=-CELT(IE)
      B(IE,3,3)=1.
      B(IE,1,4)=CELT(IE)
      B(IE,2,4)=-SELT(IE)
      B(IE,1,5)=SELT(IE)
      B(IE,2,5)=CELT(IE)
      B(IE,4,6)=1.
C     **********
C     ELEMENT S MATRICES
      S(IE,1,1)=AE/ELLEN(IE)
      S(IE,2,2)=(12.*EI)/ELLEN(IE)**3.
      S(IE,3,2)=(-6.*EI)/ELLEN(IE)**2.
      S(IE,4,2)=S(IE,3,2)
      S(IE,2,3)=S(IE,3,2)
      S(IE,3,3)=(4.*EI)/ELLEN(IE)
      S(IE,4,3)=(2.*EI)/ELLEN(IE)
      S(IE,2,4)=S(IE,3,2)
      S(IE,3,4)=S(IE,4,3)
      S(IE,4,4)=S(IE,3,3)
      DO 4 I=1,6
      DO 4 J=1,4
C     **********
C     B TRANSPOSE S ELEMENT MATRICES
      BTS(IE,I,J)=0
      DO 4 L=1,4
    4 BTS(IE,I,J)=BTS(IE,I,J)+B(IE,L,I)*S(IE,L,J)
      DO 5 I=1,6
      DO 5 J=1,6
      BTSB(I,J)=0
      DO 5 L=1,4
```

```
C     ****************************
C     ELEMENT STIFFNESS MATRICES
C     ****************************
    5 BTSB(I,J)=BTSB(I,J)+BTS(IE,I,L)*B(IE,L,J)
      DO 2 I=1,2
      MC=NBFAM(IE,I)
C     ****************************************
C     ASSEMBLE ELEMENT STIFFNESS MATRICES
C     INTO SYSTEM STIFFNESS MATRIX
C     ****************************************
      DO 2 J=1,2
      NC=NBFAM(IE,J)
      IF(NC.LT.MC) GO TO 2
      MNF=(MC-1)*NF
      NNF=(NC-1)*NF
      INF=(I-1)*NF
      JNF=(J-1)*NF
      DO 7 L=1,NF
      LK=MNF+L
      LE=INF+L
      DO 7 LL=1,NF
      LLK=NNF+LL
      IF(LLK.LT.LK) GO TO 7
      LLK=LLK-LK+1
      LLE=JNF+LL
C     *****************************************************************
C     ELEMENT STIFFNESS MATRICES ASSEMBLED INTO THE SYSTEM STIFFNESS
C     MATRIX, THE UPPER TRIANGLE IS STORED IN COLUMNS FROM DIAGONAL
C     *****************************************************************
      RK(LK,LLK)=RK(LK,LLK)+BTSB(LE,LLE)
    7 CONTINUE
    2 CONTINUE
      DO 8 I=1,NDF
      IF(IBC(I).EQ.1) GO TO 8
      DO 9 J=2,NBW
    9 RK(I,J)=0
      RK(I,1)=1
      II=I-1
      IK=2
      JJ=I-NBW+1
      IF(JJ.LT.1) JJ=1
      DO 10 IJ=II,JJ,-1
      RK(IJ,IK)=0
   10 IK=IK+1
    8 CONTINUE
C     ***********************************************************
C     SYSTEM STIFFNESS MATRIX WRITTEN OFF ONTO TEMPORARY FILE
C     ***********************************************************
      WRITE(13) ((RK(I,J),J=1,NBW),I=1,NDF)
      RETURN
      END
```

```
PLASLM.ADPF(1).STRSTR
             COMPILER (XM=3)
             SUBROUTINE STRSTR
             COMMON /EXT/RK(157,98)
      C      ***************************************************************
      C      INCREMENTS OF STRAINS AND STRESSES ARE CALCULATED. THE MOMENTS
      C      ARE SCANNED TO FIND THAT CLOSEST TO YIELDING. ALL INTERNAL FORCES,
      C      STRAINS ARE THEN FACTORED SO THAT THAT MOMENT EQUALS THE YIELD
      C      MOMENT
      C      ***************************************************************
             COMMON/BLK1/NROWP,NCOLP,MP(23,2),NF,NDF,U(73),P(73),
            #NE,IPELTS(23,4,2),PSTRN(23,4),STRN(23,4),STRS(23,4),
            #NBWA,ISIG(23,4),NELTS,LP,NB,M,ISTOP,MAX,
            #NBW,NN,E,EP,AE,EI,YSTRS,NBEAM(23,2),COORDX(23),
            #COORDY(23),ELLEN(23),SELT(23),CELT(23),NDFA,NDFS,
            #NBAS,NDFBWA,IBC(73),N1,N2,IIBC(3),PP(10,73),RLAMDA(10),
            #PPP(3),WL(3),WLOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
            #B(23,4,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALPHA(23,2),
            #DSTRN(23,4),DPSTRN(23,4),DSTRS(23,4),NELT(88,4),DU(73),
            #DP(73),RMIN,IDUM,P1(73),MS,NBS,EA(23),REI(23),IPHASE,
            #INDEX,IDUM1,LOCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
            #IEND(23),NLC,NSEC
             COMMON/BLK4/NODCNT(23)
             REAL MP
             DO 1 IE=1,NE
             DO 2 J=1,2
        2    ALPHA(IE,J)=0.
             L=0
             DO 3 I=1,2
             II=(NBEAM(IE,I)-1)*NF
             DO 3 J=1,NF
             L=L+1
        3    DEL(L)=RK(II+J,NB)
             DO 4 I=1,4
             DSTRN(IE,I)=0.
             DPSTRN(IE,I)=0.
             IIK=IPELTS(IE,1,2)
             IIJ=IPELTS(IE,2,2)
             IF(IPELTS(IE,1,1).EQ.1) DPSTRN(IE,3)=RK(IIK,NB)
             IF(IPELTS(IE,2,1).EQ.1) DPSTRN(IE,4)=RK(IIJ,NB)
             DO 4 J=1,6
      C      ******************************
      C      INCREMENT IN THE TOTAL STRAINS
      C      ******************************
        4    DSTRN(IE,I)=DSTRN(IE,I)+B(IE,I,J)*DEL(J)
             DO 25 IY=3,4
             DSTRN(IE,IY)=DSTRN(IE,IY)-DPSTRN(IE,IY)
       25    CONTINUE
             DO 6 I=1,4
             DSTRS(IE,I)=0.
             DO 6 J=1,4
      C      **********************
      C      INCREMENT IN STRESSES
      C      **********************
        6    DSTRS(IE,I)=DSTRS(IE,I)+S(IE,I,J)*DSTRN(IE,J)
      C      *****************************************
      C      CALCULATE FACTOR AND DETERMINE THE SMALLEST
      C      *****************************************
             IF(IPELTS(IE,1,1).EQ.0) ALPHA(IE,1)=ABS(MP(IE,1)/DSTRS(IE,3))
            #-STRS(IE,3)/DSTRS(IE,3)
```

```
      IF(ABS(DSTRS(IE,3)).LE.1.E-9) ALPHA(IE,1)=1.E10
      IF(IPELTS(IE,2,1).EQ.0) ALPHA(IE,2)=ABS(MP(IE,2)/DSTRS(IE,4))
     #-STRS(IE,4)/DSTRS(IE,4)
      IF(ABS(DSTRS(IE,4)).LE.1.E-9) ALPHA(IE,2)=1.E20
    1 CONTINUE
      RMIN=1.E30
      DO 7 IE=1,NE
      DO 7 J=1,2
      IF(IPELTS(IE,J,1).EQ.1) GO TO 7
      IF(ALPHA(IE,J).GT.RMIN) GO TO 7
      II=IE
      IJ=J
      RMIN=ALPHA(IE,J)
    7 CONTINUE
      NELTS=0
      DO 9 IE=1,NE
      DO 9 J=1,2
      IF(IPELTS(IE,J,1).EQ.1) GO TO 10
      IF(IE.EQ.II.AND.IJ.EQ.J) GO TO 10
      DUM=RMIN/ALPHA(IE,J)
      IF(DUM.LT.0.999) GO TO 9
   10 NELTS=NELTS+1
      NELT(NELTS,1)=IE
      NELT(NELTS,4)=J
      NODCNT(NELTS)=NBEAM(IE,J)
    9 CONTINUE
C     *****************************************************
C     MULTIPLY ALL DISPLACEMENTS,ROTATIONS,STRAINS,INTERNAL
C     FORCES,MOMENTS AND LOADS BY FACTOR CALCULATED ABOVE
C     *****************************************************
      DO 23 I=1,NDF
      DU(I)=RMIN*RK(I,NB)
      U(I)=U(I)+DU(I)
      DP(I)=RMIN*PP(LP,I)
   23 P(I)=P(I)+DP(I)
      DO 11 IE=1,NE
      DO 11 J=1,4
      DSTRN(IE,J)=DSTRN(IE,J)*RMIN
      DPSTRN(IE,J)=DPSTRN(IE,J)*RMIN
      DSTRS(IE,J)=DSTRS(IE,J)*RMIN
      STRN(IE,J)=STRN(IE,J)+DSTRN(IE,J)
      PSTRN(IE,J)=PSTRN(IE,J)+DPSTRN(IE,J)
      STRS(IE,J)=STRS(IE,J)+DSTRS(IE,J)
   11 CONTINUE
C     ******************************
C     OUTPUT RESULTS OF INCREMENTING
C     ******************************
      CALL OUTPUT
      IF(ISTOP.EQ.-1) GO TO 24
      NB=NBWA+NELTS
      M=NDF+NELTS
      IF(NB.LE.NCOLP.AND.(NDF+NELTS).LE.NROWP) GO TO 29
      PRINT 100
  100 FORMAT(1H0,26('*'),/,' AWAILABLE STORAGE EXCEEDED',
     $/,27('*'))
      STOP
C     ****************************************
C     RECOVER SYSTEM MATRIX DESTROYED IN SOLVE
C     ****************************************
   29 REWIND 13
```

```
      READ(13) ((RK(I,J),J=1,NBW),I=1,NDF)
      IF(NELTS.EQ.0) GO TO 28
      DO 14 II=1,NELTS
      NR=NDF+II
      NC=NBW+II
      DO 15 I=1,NR
   15 RK(I,NC)=0
      IE=NELT(II,1)
      IJ=NELT(II,4)
      DO 17 I=1,2
      NRR=(NBEAM(IE,I)-1)*NF
      NRE=(I-1)*NF
C     ************************************************************
C     AUGMENT SYSTEM STIFFNESS MATRIX BY ROWS AND COLUMNS
C     OF EXTENDED SYSTEM STIFFNESS MATRIX CORRESPONDING
C     TO PLASTIC HINGES
C     ************************************************************
      DO 17 J=1,NF
   17 RK(NRR+J,NC)=-BTS(IE,NRE+J,IJ+2)
      RK(NR,NC)=S(IE,IJ+2,IJ+2)
      JJ=NC+1
      DO 14 J=JJ,NB
   14 RK(NR,J)=0
   28 DO 27 I=1,NDF
C     ******************
C     ADD IN LOAD VECTOR
C     ******************
   27 RK(I,NB)=PP(LP,I)
   24 RETURN
      END
```

```
PLASLM*ADPF(1).OUTPUT
                 COMPILER (XM=3)
                 SUBROUTINE OUTPUT
                 COMMON /EXT/RK(157,98)
       C         ***********************************
       C         OUTPUTS RESULTS AFTER EACH INCREMENT
       C         ***********************************
                 COMMON/BLK1/NROWP,NCOLP,MP(23,2),NF,NDF,U(73),P(73),
                #NE,IPELTS(23,4,2),PSTRN(23,4),STRN(23,4),STRS(23,4),
                #NBWA,ISIG(23,4),NELTS,LP,NB,M,ISTOP,MAX,
                #NBW,NU,E,EP,AE,EI,YSTRS,NBEAM(23,2),COORDX(23),
                #COORDY(23),ELLEN(23),SELT(23),CELT(23),NDFA,NDFS,
                #NBWS,NDFBWA,IBC(73),N1,N2,IIBC(3),PP(10,73),RLAMDA(10),
                #PPP(3),NL(3),NLOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
                #B(23,4,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALPHA(23,2),
                #DSTRN(23,4),DPSTRN(23,4),DSTRS(23,4),NELT(88,4),DU(73),
                #DP(73),RMIN,IDUM,P1(73),MS,NBS,EA(23),REI(23),IPHASE,
                #INDEX,IDUM1,LDCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
                #IEND(23),NLC,NSEC
                 COMMON/BLK3/RLDFCT(30),NL1,DLDFCT(30),ALDFCT(30)
                 COMMON/BLK4/NODCNT(23)
                 COMMON/BLK9/IBP,ISHORT,ITAB
                 IF(DET.NE.0) GO TO 501
                 IF(IDUM.EQ.-1) GO TO 500
       501       REAL MP
                 NL1=NL1+1
                 IF(NL1.EQ.1) NOUT=2
                  IF(ISHORT.NE.1)PRINT 200,NL1
       200       FORMAT(1H1,'RESULTS AFTER',I4,2X,'LOAD INCREMENTS',/)
                 NOUT=NOUT+1
                 IF(ISHORT.NE.1)PRINT 201,(NELT(IE,1),IE=1,NELTS)
          201    FORMAT(1H0,'NEXT ELEMENTS TO UNDERGO PLASTIC DEFORMATION :',10I5)
                 IF(ISHORT.NE.1)PRINT 202,(NODCNT(IE),IE=1,NELTS)
       202       FORMAT(1H ,'AT NODES : ',10I5)
               2 IF(ISHORT.NE.1)PRINT 203,RMIN
          203    FORMAT(1H ,'AT LOAD FACTOR',E11.6,/)
                 DLDFCT(NL1)=RMIN
                 RLDFCT(LP)=RLDFCT(LP)+RMIN
                 IF(ISHORT.NE.1)PRINT 301
       301       FORMAT(1H ,///26X,'INCREMENT IN LOADS',28X,'TOTAL LOADS',
                #/,10X,'NODE',6X,'DPX',10X,'DPY',10X,'DMOM',14X,'PX',
                #11X,'PY',10X,'MOM',/)
                 DO 333 I=1,NN
                 NR=(I-1)*NF
                 J=NR+1
                 K=NR+2
                 L=NR+3
       C         ********************************
       C         INCREMENTS IN LOADS                        TOTAL LOADS
       C         ********************************
                 IF(ISHORT.NE.1)PRINT 302,I,DP(J),DP(K),DP(L),P(J),P(K),P(L)
       302       FORMAT(1H ,11X,I2,4X,3(E10.5,3X),4X,3(E10.5,3X))
       333       CONTINUE
                 IF(ISHORT.NE.1)PRINT 334
       C         ***********************************************
       C         INCREMENTS IN DISPLACEMENTS        TOTAL DISPLACEMENTS
       C         ***********************************************
       334       FORMAT(1H ,///,20X,'INCREMENT IN DISPLACEMENTS',25X,
                #'TOTAL DISPLACEMENTS',/,10X,'NODE',7X,'DU',11X,'DV',10X,
                #'DTHETA',12X,'U',12X,'V',11X,'THETA',/)
```

```
      DO 335 I=1,NN
      NR=(I-1)*NF
      J=NR+1
      K=NR+2
      L=NR+3
      IF(ISHORT.NE.1)PRINT 336,I,DU(J),DU(K),DU(L),U(J),U(K),U(L)
  336 FORMAT(1H ,11X,I2,4X,3(E10.5,3X),4X,3(E10.5,3X))
  335 CONTINUE
      IF(ISHORT.NE.1)PRINT 303
  303 FORMAT(1H ,///,24X,'INCREMENT IN STRAINS',42X,'TOTAL STRAINS'
     #,/,1X,'ELEMENT',6X,'AXIAL',8X,'SHEAR',7X,'DTHETA(A)',
     #4X,'DTHETA(B)',11X,'AXIAL',8X,'SHEAR',6X,'THETA(A)',
     #5X,'THETA(B)',/)
C     ******************************************
C     INCREMENTS IN STRAINS                TOTAL STRAINS
C     ******************************************
      DO 337 IN=1,NE
      IF(ISHORT.NE.1)PRINT 304,(IN,(DSTRN(IN,I1),I1=1,4),
     #(STRN(IN,I2),I2=1,4))
  304 FORMAT(1H ,3X,I2,6X,4(E10.5,3X),6X,4(E10.5,3X),/)
  337 CONTINUE
      IF(ISHORT.NE.1)PRINT 338
  338 FORMAT(1H ,///,56X,'PLASTIC STRAINS',/,41X,'ELEMENT',
     #4X,'THETAP(A)',6X,'THETAP(B)')
      DO 340 IE=1,NE
      IF(ISHORT.NE.1)PRINT 339,IE,(PSTRN(IE,I1),I1=3,4)
  339 FORMAT(1H ,43X,I2,5X,E10.5,5X,E10.5)
  340 CONTINUE
      IF(ISHORT.NE.1)PRINT 305
C     *******************************
C     INCREMENTS IN INTERNAL FORCES
C     *******************************
  305 FORMAT(1H ,///,40X,'INCREMENT IN INTERNAL NODE FORCES',/,
     #8X,'ELEMENT',7X,'AXIAL',14X,'SHEAR',18X,'BENDING(A)',
     #10X,'BENDING(B)',/)
      DO 345 IE=1,NE
      IF(ISHORT.NE.1)PRINT 306,(IE,(DSTRS(IE,I),I=1,4))
  306 FORMAT(1H ,10X,I2,7X,E10.5,10X,E10.5,15X,E10.5,10X,E10.5)
  345 CONTINUE
      IF(ISHORT.NE.1)PRINT 307
C     ***********************
C     TOTAL INTERNAL FORCES
C     ***********************
  307 FORMAT(1H ,///,44X,'TOTAL INTERNAL NODE FORCES',/,2X,
     #'ELEMENT',6X,'AXIAL',7X,'SHEAR',6X,'BENDING(A)',4X,
     #'YIELD MOMA',4X,'BENDING(B)',4X,'YIELD MOMB',/)
      DO 945 IE=1,NE
      IF(ISHORT.NE.1)PRINT 308,IE,STRS(IE,1),STRS(IE,2),
     #STRS(IE,3),MP(IE,1),STRS(IE,4),MP(IE,2)
  308 FORMAT(1H ,4X,I2,6X,6(E10.5,4X))
  945 CONTINUE
      DO 349 I=1,NUF
      DO 349 IE=1,NE
      IF(ABS(STRS(IE,3))/MP(IE,1).GT.1.001) GO TO 280
      IF(ABS(U(I)).GT.1) GO TO 280
      IF(ABS(STRS(IE,4))/MP(IE,2).GT.1.001) GO TO 280
  349 CONTINUE
      GO TO 100
  280 ALDFCT(LP)=RLDFCT(LP)
      LP=LP+1
```

```
      C     *********************************************
      C     IF A MECHANISM HAS FORMED COMMENCE NEXT LOAD CASE
      C     *********************************************
            IF(ISHORT.NE.1)PRINT 279
      279   FORMAT(1H ,///,56X,16('*'),/,56X,'MECHANISM FORMED',/,
           $56X,16('*'))
            IF(ISHORT.NE.1)PRINT 2799,NL1
      2799  FORMAT(1H ,//,13X,90('*'),/,13X,' THE RESULTS OF'
           #' THE ABOVE ITERATION',I3,' ARE MEANINGLESS DUE'
           #' TO THE FORMATION OF A MECHANISM',/,13X,90('*'))
      C     *********************************
      C     IF ALL LOAD CASES HAVE BEEN COMPLETED
      C     CHOOSE ADEQUATE SECTIONS FOR MEMBERS
      C     *********************************
      500   IF(LP.GT.NLC) CALL SORT
            IF(ISHORT.NE.1)PRINT 400,LP
      400   FORMAT(1H1,///,56X,' LOAD CASE',2X,I2)
            DET=0
            IDUM=-1
            DO 401 I=1,NDF
            U(I)=0.
            P1(I)=P(I)
      C     *********************
      C     INITIALISE VARIABLES
      C     *********************
      401   P(I)=0.
            DO 402 IE=1,NE
            DO 402 J=1,4
            DO 402 I=1,2
            DSTRN(IE,J)=0
            DSTRS(IE,J)=0
            IPELTS(IE,I,J)=0
            PSTRN(IE,J)=0
            STRN(IE,J)=0
            STRS(IE,J)=0
      402   ISIG(IE,J)=0.
            NELTS=0
            NL1=0
            NB=NBWA
            M=NDF
            DO 403 I=1,NDF
      403   RK(I,NB)=PP(LP,I)
      404   CALL SOLVE
      405   CALL STRSTR
            IF(NELTS.EQ.0) GO TO 404
            CALL ITER
            GO TO 405
      100   RETURN
            END
```

```
PLASLM*ADPF(1).ITER
      COMPILER (XM=3)
      SUBROUTINE ITER
      COMMON /EXT/RK(157,98)
C     *******************
C     ITERATION PROCEDURE
C     *******************
      COMMON/BLK1/NROWP,NCOLP,MP(23,2),NF,NDF,U(73),P(73),
     #NE,IPELTS(23,4,2),PSTRN(23,4),STRN(23,4),STRS(23,4),
     #NBWA,ISIG(23,4),NELTS,LP,NB,M,ISTOP,MAX,
     #NBW,NN,E,EP,AE,EI,YSTRS,NBEAM(23,2),COORDX(23),
     #COORDY(23),ELLEN(23),SELT(23),CELT(23),NDFA,NDFS,
     #NBWS,NDFBWA,IBC(73),N1,N2,IIBC(3),PP(10,73),RLAMDA(10),
     #PPP(3),WL(3),WLOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
     #B(23,4,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALPHA(23,2),
     #DSTRN(23,4),DPSTRN(23,4),DSTRS(23,4),NELT(88,4),DU(73),
     #DP(73),RMIN,IDUM,P1(73),MS,NBS,EA(23),REI(23),IPHASE,
     #INDEX,IDUM1,LDCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
     #IEND(23),NLC,NSEC
      COMMON/BLK3/RLDFCT(30),NL1,DLDFCT(30),ALDFCT(30)
      COMMON/BLK9/IBP,ISHORT,ITAB
C     ************************************************
C     ROWS AND COLUMNS OF THE EXTENDED STIFFNESS MATRIX
C     ARE WRITTEN OFF ONTO TEMPORARY FILE
C     ************************************************
      REWIND 14
      WRITE(14) ((RK(I,J),J=NBWA,NB),I=1,M)
      DO 2 I=1,NELTS
    2 NELT(I,3)=1
   16 CALL SOLVE
      DO 3 I=1,NELTS
    3 NELT(I,2)=1
      DO 4 IR=NDFA,M
      II=IR-NDF
      IE=NELT(II,1)
      J=NELT(II,4)
      IF(ABS(RK(IR,NB)).LT.1.E-04) GO TO 4
C     *********************************************
C     CHECK THAT PLASTIC ROTATIONS AND MOMENTS ARE
C     OF THE SAME SIGN
C     *********************************************
      IF((RK(IR,NB)*STRS(IE,J+2)).GE.0.) GO TO 4
      NELT(II,2)=0
    4 CONTINUE
      DO 10 IIE=1,NELTS
   10 IF(NELT(IIE,2).EQ.0) GO TO 11
      GO TO 12
   11 REWIND 13
      READ(13) ((RK(I,J),J=1,NBW),I=1,NDF)
      REWIND 14
      READ(14) ((RK(I,J),J=NBWA,NB),I=1,M)
      DO 14 IE=1,NELTS
      IF(NELT(IE,2).EQ.1) GO TO 14
      J=NBW+IE
      DO 15 I=1,M
   15 RK(I,J)=0
      RK(NDF+IE,J)=1
      NELT(IE,3)=0
   14 CONTINUE
      GO TO 16
```

```
      12 DO 17 IE=1,NE
         DO 17 J=1,2
      17 IPELTS(IE,J,1)=0
         IF((OUT(ISTOP)-OUT(ISTOP-1)).GT.0) GO TO 19
         DO 20 II=1,ISTOP
   20    OUT(II)=0.
         ALDFCT(LP)=RLDFCT(LP)
         IF(NL1.EQ.1) ALDFCT(LP)=DLDFCT(NL1)
         LP=LP+1
         IDUM=-1
         DET=0
         ISTOP=0
         IF(ISHORT.NE.1)PRINT 702
  702    FORMAT(1H ,///,57X,21('*'),/,57X,
        #'INSTABILITY INDICATED',/,57X,21('*'))
         CALL OUTPUT
   19    DO 18 I=1,NELTS
         IF(NELT(I,3).EQ.0.AND.ISHORT.NE.1) PRINT 702
C        **********
C        UNLOADING
C        **********
          IF(NELT(I,3).EQ.0) ALDFCT(LP)=RLDFCT(LP)
         IF(NELT(I,3).EQ.0) LP=LP+1
         IF(NELT(I,3).EQ.0) IDUM=-1
         IF(NELT(I,3).EQ.0) DET=0
         IF(LP.GT.NLC) CALL SORT
         IF(NELT(I,3).EQ.0) CALL OUTPUT
         IE=NELT(I,1)
         J=NELT(I,4)
         IPELTS(IE,J,1)=1
         IPELTS(IE,J,2)=NDF+1
      18 CONTINUE
         RETURN
         END

@PRT,S ADPF.SOLVED,.ELTHAD,.STRSTD,.OUTPUD,.ITED
```

```
PLASLM*ADPF(1).SOLVED
          COMPILER (XM=3)
          SUBROUTINE SOLVED
          COMMON /EXT/RK(157,98)
    C     ***************************************************************
    C     SOLUTION ROUTINE FOR DEFORMATION ANALYSIS  :  GAUSS-JORDAN REDUCTION
    C     ***************************************************************
          COMMON/BLK1/NROWP,NCOLP,MP(23,2),NF,NDF,U(73),P(73),
         #NE,IPELTS(23,4,2),PSTRN(23,4),STRN(23,4),STRS(23,4),
         #NBWA,ISIG(23,4),NELTS,LP,NB,M,ISTOP,MAX,
         #NBW,NN,E,EP,AE,EI,YSTRS,NBEAM(23,2),COORDX(23),
         #COORDY(23),ELLEN(23),SELT(23),CELT(23),NDFA,NDFS,
         #NBWS,NDFBWA,IBC(73),N1,N2,IIBC(3),PP(10,73),RLAMDA(10),
         #PPP(3),WL(3),WLOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
         #B(23,4,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALPHA(23,2),
         #DSTRN(23,4),DPSTRN(23,4),DSTRS(23,4),NELT(88,4),DU(73),
         #DP(73),RMIN,IDUM,P1(73),MS,NBS,EA(23),RCI(23),IPHASE,
         #INDEX,IDUM1,LDCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
         #IEND(23),NLC,NSEC
          REWIND 15
          READ(15) ((RK(I,J),J=1,NB),I=1,M)
          DO 33 I=1,NDF
       33 RK(I,NB)=WLOAD(LDCASE,I)
          REAL MP
          MS=M-1
          NBS=NB-1
          NBWS=NBW-1
          NDFBWA=NDF-NBWS+1
          KK=NBW
          K=NDF
          DO 900 IE=1,NE
          DO 900 J=1,4
          KK=KK+1
          K=K+1
          IF(ISIG(IE,J).NE.0) GO TO 901
          DO 902 L=KK,NB
      902 RK(K,L)=0.
          DO 903 L=1,K
      903 RK(L,KK)=0.
          RK(K,KK)=1.
          GO TO 900
      901 RK(K,NB)=-ISIG(IE,J)*MP(IE,J-2)
      900 CONTINUE
          DET=1.
          DO 1 IR=1,NDF
          IF(IR.EQ.NDF) GO TO 10
          IB=IR+1
          IBB=IR+NBWS
          IF(IBB.GT.NDF) IBB=NDF
          NCOLL=NBWS
          IC=2
          DO 8 I=IB,IBB
          ICC=IC
          IF(RK(IR,IC).EQ.0) GO TO 20
          FACT=RK(IR,IC)/RK(IR,1)
          IF(IR.GE.NDFBWA) NCOLL=NDFA-1
          DO 9 J=1,NCOLL
          RK(I,J)=RK(I,J)-FACT*RK(IR,ICC)
        9 ICC=ICC+1
          DO 16 J=NBWA,NB
```

```
   16 RK(I,J)=RK(I,J)-FACT*RK(IR,J)
   20 NCOLL=NCOLL-1
    8 IC=IC+1
   10 DET=DET*(RK(IR,1))
      IF(NELTS.EQ.0) GO TO 1
      NCOLF=NBWA
      DO 15 I=NDFA,M
      IF(RK(IR,NCOLF).EQ.0) GO TO 15
      FACT=RK(IR,NCOLF)/RK(IR,1)
      DO 17 J=NCOLF,NB
   17 RK(I,J)=RK(I,J)-FACT*RK(IR,J)
   15 NCOLF=NCOLF+1
    1 CONTINUE
      IF(NELTS.EQ.0) GO TO 21
      IF(NELTS.EQ.1) GO TO 11
      NCOLF=NBWA
      DO 2 IR=NDFA,MS
      IRA=IR+1
      IC=NCOLF+1
      DO 3 I=IRA,M
      IF(RK(IR,IC).EQ.0) GO TO 3
      FACT=RK(IR,IC)/RK(IR,NCOLF)
      DO 4 J=IC,NB
    4 RK(I,J)=RK(I,J)-FACT*RK(IR,J)
    3 IC=IC+1
      DET=DET*(RK(IR,NCOLF))
    2 NCOLF=NCOLF+1
   11 CONTINUE
      RK(M,NB)=RK(M,NB)/RK(M,NBS)
      DET=DET*(RK(M,NBS))
      IF(NELTS.EQ.1) GO TO 7
      NCOLF=NBS
      DO 5 IR=MS,NDFA,-1
      II=IR+1
      DO 6 J=NCOLF,NBS
      RK(IR,NB)=RK(IR,NB)-RK(IR,J)*RK(II,NB)
    6 II=II+1
      NCOLF=NCOLF-1
    5 RK(IR,NB)=RK(IR,NB)/RK(IR,NCOLF)
    7 II=NDFA
      DO 12 J=NBWA,NBS
      DO 13 I=1,NDF
   13 RK(I,NB)=RK(I,NB)-RK(I,J)*RK(II,NB)
   12 II=II+1
   21 RK(NDF,NB)=RK(NDF,NB)/RK(NDF,1)
      DO 14 IR=NDFS,1,-1
      II=IR+1
      NCOLL=NBW
      IF(IR.GE.NDFBWA) NCOLL=NDFA-IR
      DO 19 J=2,NCOLL
      RK(IR,NB)=RK(IR,NB)-RK(IR,J)*RK(II,NB)
   19 II=II+1
   14 RK(IR,NB)=RK(IR,NB)/RK(IR,1)
      RETURN
      END
```

```
PLASLH*ADPF(1).ELTMAD
      COMPILER (XM=3)
      SUBROUTINE ELTMAD
      COMMON /EXT/RK(157,98)
C     ******************************************************************
C     SETS UP DEFORMATION AND STIFFNESS MATRICES FOR EACH ELEMENT
C     ******************************************************************
      COMMON/BLK1/NROWP,NCOLP,MP(23,2),NF,NDF,U(73),P(73),
     #NE,IPELTS(23,4,2),PSTRN(23,4),STRN(23,4),STRS(23,4),
     #NBWA,ISIG(23,4),NELTS,LP,NB,M,ISTOP,MAX,
     #NBW,NN,E,EP,AE,EI,YSTRS,NBEAM(23,2),COORDX(23),
     #COORDY(23),ELLEN(23),SELT(23),CELT(23),NDFA,NDFS,
     #NBWS,NDFBWA,IBC(73),N1,N2,IIBC(3),PP(10,73),RLAMDA(10),
     #PPP(3),WL(3),WLOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
     #B(23,4,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALBHA(23,2),
     #DSTRN(23,4),DPSTRN(23,4),DSTRS(23,4),NELT(88,4),DU(73),
     #DP(73),RMIN,IDUM,P1(73),MS,NBS,EA(23),REI(23),IPHASE,
     #INDEX,IDUM1,LDCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
     #IEND(23),NLC,NSEC
      COMMON/BLK2/CSTRS(10,23,4),EFFLEN(23),ICOL(10),
     #CM1(23),CM(23),PE(23),BUKCHK(23,2),BM(23),RMM(23),
     #DMP(23),RETURN(10,23)
      COMMON/BLK6/SSP(23,20)
      COMMON/BLK9/IBP,ISHORT,ITAB
      REAL MP
      DO 333 I=1,M
      DO 333 J=1,NB
  333 RK(I,J)=0.
      DO 2 IE=1,NE
      DO 1 I1=1,4
      DO 3 I2=1,4
    3 S(IE,I1,I2)=0.
C     **********
      DO 1 I3=1,6
    1 B(IE,I1,I3)=0.
      B(IE,1,1)=-CELT(IE)
      B(IE,1,2)=-SELT(IE)
      B(IE,2,1)=SELT(IE)
      B(IE,2,2)=-CELT(IE)
      B(IE,3,3)=1.
      B(IE,1,4)=CELT(IE)
      B(IE,2,4)=-SELT(IE)
      B(IE,1,5)=SELT(IE)
      B(IE,2,5)=CELT(IE)
      B(IE,4,6)=1.
C     **********
      S(IE,1,1)=EA(IC)/ELLEN(IE)
      S(IE,2,2)=(12.*REI(IE))/ELLEN(IE)**3.
      S(IE,3,2)=(-6.*REI(IE))/ELLEN(IE)**2,
      S(IE,4,2)=S(IE,3,2)
      S(IE,2,3)=S(IE,3,2)
      S(IE,3,3)=(4.*REI(IE))/ELLEN(IE)
      S(IE,4,3)=(2.*REI(IE))/ELLEN(IE)
      S(IE,2,4)=S(IE,3,2)
      S(IE,3,4)=S(IE,4,3)
      S(IE,4,4)=S(IC,3,3)
C     **********
      DO 4 I=1,6
      DO 4 J=1,4
      BTS(IE,I,J)=0
```

```
      DO 4 L=1,4
    4 BTS(IE,I,J)=BTS(IE,I,J)+B(IE,L,I)*S(IE,L,J)
      DO 5 I=1,6
      DO 5 J=1,6
      BTSB(I,J)=0
      DO 5 L=1,4
    5 BTSB(I,J)=BTSB(I,J)+BTS(IE,I,L)*B(IE,L,J)
      DO 2 I=1,2
      MC=NBEAM(IE,I)
      DO 2 J=1,2
      NC=NBEAM(IE,J)
      IF(NC.LT.MC) GO TO 2
      MNF=(MC-1)*NF
      NNF=(NC-1)*NF
      INF=(I-1)*NF
      JNF=(J-1)*NF
      DO 7 L=1,NF
      LK=MNF+L
      LE=INF+L
      DO 7 LL=1,NF
      LLK=NNF+LL
      IF(LLK.LT.LK) GO TO 7
      LLK=LLK-LK+1
      LLE=JNF+LL
C     **************************************************************
C     ELEMENT STIFFNESS MATRICES ASSEMBLED INTO THE SYSTEM STIFFNESS
C     MATRIX. THE UPPER TRIANGLE IS STORED IN COLUMNS FROM DIAGONAL
C     **************************************************************
      RK(LK,LLK)=RK(LK,LLK)+BTSB(LE,LLE)
    7 CONTINUE
    2 CONTINUE
      DO 6 IE=1,NE
      IC=NBW+(IE-1)*4
      IIEND=1
      IR=(NBEAM(IE,IIEND)-1)*NF
      DO 231 I=1,NF
      DO 231 J=1,4
  231 RK(IR+I,IC+J)=-BTS(IE,I,J)
      IIEND=2
      IR=(NBEAM(IE,IIEND)-1)*NF
      DO 232 II=4,6
      DO 232 JJ=1,4
  232 RK(IR+II-3,IC+JJ)=-BTS(IE,II,JJ)
      IR=NDF+(IE-1)*4
      DO 347 I=1,4
      DO 347 J=1,4
  347 RK(IR+I,IC+J)=S(IE,I,J)
    6 CONTINUE
      N=NBW+4*NE+1
      DO 8 I=1,NDF
      IF(IBC(I).EQ.1) GO TO 8
      DO 9 J=2,N
    9 RK(I,J)=0.
      RK(I,1)=1.
      II=I-1
      IK=2
      JJ=I-NBW+1
      IF(JJ.LT.1) JJ=1
      DO 10 IJ=II,JJ,-1
      RK(IJ,IK)=0.
```

```
   10   IK=IK+1
   8    CONTINUE
        REWIND 15
        WRITE(15) ((RK(I,J),J=1,NB),I=1,M)
        RETURN
        END
```

```
PLASLM*ADPF(1).STRSTD
      C     ************************************************************
      C     STRAINS AND INTERNAL FORCES ARE CALCULATED AFTER EACH ITERATION
      C     ************************************************************
            COMPILER (XM=3)
            SUBROUTINE STRSTD
            COMMON /EXT/RK(157,96)
            COMMON/BLK1/NROWP,NCOLP,MP(23,2),NF,NDF,U(73),P(73),
           #NE,IPELTS(23,4,2),PSTRN(23,4),STRN(23,4),STRS(23,4),
           #NBWA,ISIG(23,4),NELTS,LP,NB,M,ISTOP,MAX,
           #NBW,NM,E,EP,AE,EI,YSTRS,NBEAM(23,2),COORDX(23),
           #COORDY(23),ELLEN(23),SELT(23),CELT(23),NDFA,NDFS,
           #NBWS,NDFBWA,IBC(73),N1,N2,IIBC(3),PP(10,73),RLAMDA(10),
           #PPP(3),WL(3),WLOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
           #B(23,4,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALPHA(23,2),
           #DSTRN(23,4),DPSTRN(23,4),DSTRS(23,4),NELT(88,4),DU(73),
           #DP(73),RMIN,IDUM,P1(73),MS,NBS,EA(23),RE1(23),IPHASE,
           #INDEX,IDUM1,LUCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
           #IEND(23),NLC,USEC
            COMMON/BLK2/CSTRS(10,23,4),EFFLEN(23),ICOL(10),
           #CM1(23),CM(23),PE(23),BUKCHK(23,2),BM(23),RMM(23),
           #DMP(23),RETURN(10,23)
            COMMON/BLK7/LCRTCS
            COMMON/BLK8/DEF(10,73),DEFCHU(20,3),NCHU,NCH,DEFCH(20,3)
            REAL MP
            IDUM=0
            DO 7 I=1,NDF
        7   U(I)=RK(I,NB)
            DO 1 IE=1,NE
            L=0
            DO 2 J=1,4
        2   PSTRN(IE,J)=0.
            JJI=4*IE+NDF-1
            K=4*IE+NDF
            PSTRN(IE,3)=RK(JJI,NB)
            PSTRN(IE,4)=RK(K,NB)
            DO 3 I=1,2
            II=(NBEAM(IE,I)-1)*NF
            DO 3 J=1,NF
            L=L+1
        3   DEL(L)=RK(II+J,NB)
            DO 4 I=1,4
            STRN(IE,I)=0.
            DO 4 J=1,6
      C     ********
      C     STRAINS
      C     ********
        4   STRN(IE,I)=STRN(IE,I)+B(IE,I,J)*DEL(J)
            DO 66 IY=3,4
       66   STRN(IE,IY)=STRN(IE,IY)-PSTRN(IE,IY)
            DO 6 I=1,4
            STRS(IE,I)=0.
            DO 6 J=1,4
      C     ***************
      C     INTERNAL FORCES
      C     ***************
        6   STRS(IE,I)=STRS(IE,I)+S(IE,I,J)*STRN(IE,J)
        1   CONTINUE
            IF(INDEX.EQ.1) CALL OUTPUD
      C     ******************************
```

```
C     OUTPUT RESULTS OF ITERATION
C     ******************************
      DO 11 IE=1,NE
      DO 11 I=3,4
      II=I-2
      SA=ABS(STRS(IE,I))/MP(IE,II)
C     ******************************************
C     CHECK THAN YIELD MOMENTS ARE NOT EXCEEDED
C     ******************************************
      IF(SA.LE.0.99999) GO TO 12
      IF(SA.GE.0.9999.AND.SA.LE.1.0001) GO TO 11
      IF(ABS(STRS(IE,I))/MP(IE,II).GT.0.9999) CALL ITED
12    IF((STRS(IE,I)*PSTRN(IE,I)).LT.0.0) CALL ITED
11    CONTINUE
      IF(INDEX.EQ.1) IDUM=-1
      CALL OUTPUD
      IF(NCH.EQ.0) GO TO 8
      PRINT 13
C     *****************
C     DEFLECTION CHECKS
C     *****************
13    FORMAT(1H ,///,41X,'COMMENCE DEFLECTION CHECKS',/,41X,26('*'))
      PRINT 400
400   FORMAT(1H ,/,45X,'AT ULTIMATE LOADS')
      DO 9 I=1,NCHU
      J=(DEFCHU(I,1)-1)*NF+DEFCHU(I,2)
      JJ=DEFCHU(I,1)
      PRINT 10,JJ,DEF(IPHASE,J),DEFCHU(I,3)
10    FORMAT(1H ,//5X,'NODE ',I2,/,5X,
     #'DEFLECTION IS ',E11.6,10X,' PERMISSIBLE :',1X,E11.6)
      IF((ABS(DEF(IPHASE,J))-DEFCHU(I,3)).GT.0) PRINT 14
14    FORMAT(1H ,'   EXCESSIVE DEFLECTIONS : STIFFEN STRUCTURE')
      IF((ABS(DEF(IPHASE,J))-DEFCHU(I,3)).GT.0) STOP
9     CONTINUE
      DO 15 IE=1,NE
      DO 15 I=1,4
      CSTRS(IPHASE,IE,I)=STRS(IE,I)
15    CONTINUE
8     IPHASE=IPHASE+1
      IF(IPHASE.GT.NLC) CALL CHECK
      CALL CONTRL
      END
```

```
PLASLH*ADPF(1).OUTPUD
            COMPILER (XM=3)
            SUBROUTINE OUTPUD
            COMMON /EXT/RK(157,98)
      C     *******************************************************.
      C     OUTPUTS RESULTS OF ANALYSIS AT WORKING AND FACTORED LOADS
      C     *******************************************************.
            COMMON/BLK1/NROWP,NCOLP,MP(23,2),NF,NDF,U(73),P(73),
           #NE,IPELTS(23,4,2),PSTRN(23,4),STRN(23,4),STRS(23,4),
           #NBWA,ISIG(23,4),NELTS,LP,NB,M,ISTOP,MAX,
           #NBW,NN,E,EP,AE,EI,YSTRS,NBEAM(23,2),COORDX(23),
           #COORDY(23),ELLEN(23),SELT(23),CELT(23),NDFA,NDFS,
           #NBWS,NDFBWA,IBC(73),N1,N2,IIBC(3),PP(10,73),RLAMDA(10),
           #PPP(3),WL(3),WLOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
           #B(23,4,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALPHA(23,2),
           #DSTRN(23,4),DPSTRN(23,4),DSTRS(23,4),NELT(88,4),DU(73),
           #DP(73),RMIN,IDUM,P1(73),MS,NBS,EA(23),REI(23),IPHASE,
           #INDEX,IDUM1,LDCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
           #IEND(23),NLC,NSEC
            COMMON/BLK2/CSTRS(10,23,4),EFFLEN(23),ICOL(10),
           #CM1(23),CM(23),PE(23),BUKCHK(23,2),BM(23),RMM(23),
           #DMP(23),RETURN(10,23)
            COMMON/BLK8/DEF(10,73),DEFCHU(20,3),NCHU,NCH,DEFCH(20,3)
            COMMON/BLK9/IBP,ISHORT,ITAB
            REAL MP
            IF(INDEX.EQ.1.AND.IBP.EQ.1.AND.IDUM.NE.-1) PRINT 22,IPHASE
         22 FORMAT(1H1,' LOAD CASE',1X,I2)
            IF(INDEX.GT.1) GO TO 16
            IF(IDUM.EQ.-1) GO TO 16
            PRINT 21
         21 FORMAT(1H ,///,47X,' SOLUTION AT WORKING LOADS')
            DO 17 IE=1,NE
            DO 17 IJ=1,4
            STRN(IE,IJ)=STRN(IE,IJ)/RLAMDA(IPHASE)
            STRS(IE,IJ)=STRS(IE,IJ)/RLAMDA(IPHASE)
         17 CONTINUE
            DO 166 I=1,NDF
            U(I)=U(I)/RLAMDA(IPHASE)
        166 CONTINUE
            DO 12 IE=1,NE
            DO 12 IJ=3,4
         12 STRN(IE,IJ)=STRN(IE,IJ)+PSTRN(IE,IJ)
         16 CONTINUE
            IF(IDUM.EQ.-1)PRINT 18
            IF(INDEX.GT.1) PRINT 18
         18 FORMAT(1H1,///,46X,' SOLUTION AT FACTORED LOADS')
            PRINT 1,INDEX
          1 FORMAT(1H ,'RESULTS AFTER',I3,2X,'ITERATIONS')
            PRINT 6
          6 FORMAT(1H ,52X,'NODE DISPLACEMENTS',//,32X,
           #'NODE',10X,'DX',14X,'DY',11X,'ROTATION',/)
            DO 7 I=1,NN
            NR=(I-1)*NF
            J=NR+1
            L=NR+2
            K=NR+3
            PRINT 8,(I,U(J),U(L),U(K))
          8 FORMAT(1H ,33X,I3,6X,E10.5,2(5X,E10.5))
          7 CONTINUE
            IF(ISHORT.NE.1) PRINT 2
```

```
    2 FORMAT(1H ,/,51X,'MEMBER DEFORMATIONS',//22X,
     #'ELEMENT',7X,'AXIAL',8X,'TRANSVERSE',6X,'THETA(A)',
     #7X,'THETA(B)',/)
      IF(ISHORT.NE.1) PRINT 3,((IE,STRN(IE,1),STRN(IE,2)
     #,STRN(IE,3),STRN(IE,4)),IE=1,NE)
    3 FORMAT(1H ,24X,I3,6X,E10.5,5X,E10.5,6X,E10.5,
     #5X,E10.5)
    5 FORMAT(1H ,10X,I3,6X,E10.5,5X,E10.5,6X,E10.5,
     #5X,E10.5,5X,E10.5,4X,E10.5)
      PRINT 4
    4 FORMAT(1H ,/,50X,'INTERNAL NODE FORCES',//,10X,
     #'ELEMENT',7X,'AXIAL',10X,'SHEAR',7X,'BENDING(A)',
     #5X,'YIELD MOM(A)',3X,'BENDING(B)',4X,'YIELD MOM(B)',/)
      PRINT 5,((IE,STRS(IE,1),STRS(IE,2),STRS(IE,3),MP(IE,1),
     #STRS(IE,4),MP(IE,2)),IE=1,NE)
      IF(INDEX.GT.1) GO TO 768
      DO 932 I=1,NDF
  932 DEF(IPHASE,I)=U(I)
      IF(NCH.EQ.0) GO TO 88
      IF(IDUM.EQ.-1) GO TO 88
      PRINT 133
  133 FORMAT(1H ,///,41X,'COMMENCE DEFLECTION CHECKS',/,41X,26('*'))
      PRINT 440
  440 FORMAT(1H ,/,46X,'AT WORKING LOADS')
      DO 937 I=1,NCH
      J=(DEFCH(I,1)-1)*NF+DEFCH(I,2)
      JJ=DEFCH(I,1)
      PRINT 1076,JJ,DEF(IPHASE,J),DEFCH(I,3)
 1076 FORMAT(1H ,//5X,'NODE ',I2,/,5X,
     #'DEFLECTION IS ',E11.6,10X,' PERMISSIBLE :',1X,E11.6)
      IF((ABS(DEF(IPHASE,J))-DEFCH(I,3)).GT.0) PRINT 155
  155 FORMAT(1H ,'   EXCESSIVE DEFLECTIONS : STIFFEN STRUCTURE')
  937 CONTINUE
   88 IF(INDEX.EQ.1) GO TO 19
  768 PRINT 9
    9 FORMAT(1H ,/,53X,'PLASTIC ROTATIONS',/,31X,'ELEMENT',
     #3X,'NODE END',10X,'THETA(P)',/)
      II=0
  193 II=II+1
      IF(II.GT.NE) GO TO 199
      IF(ABS(PSTRN(II,3)).LE.1.E-8.AND.ABS(PSTRN(II,4)).
     #LE.1.E-8) GO TO 193
      PRINT 10,(NBEAM(II,1),PSTRN(II,3),II,NBEAM(II,2),PSTRN(II,4))
   10 FORMAT(1H ,44X,I3,11X,E10.5,/,33X,I3,/,45X,
     #I3,11X,E10.5,//)
      GO TO 193
  199 IF(INDEX.GT.1) GO TO 25
   19 IF(IDUM.EQ.-1) GO TO 25
      DO 20 IE=1,NE
      DO 20 IJ=1,4
      STRN(IE,IJ)=STRN(IE,IJ)*RLAMDA(IPHASE)
      STRS(IE,IJ)=STRS(IE,IJ)*RLAMDA(IPHASE)
   20 CONTINUE
      DO 255 I=1,NDF
      U(I)=U(I)*RLAMDA(IPHASE)
  255 CONTINUE
   25 CONTINUE
      IDUM=0
      DO 15 IE=1,NE
      DO 15 I=1,4
```

```
15   CSTRS(IPHASE,IE,I)=STRS(IE,I)
     DO 14 I=1,NDF
14   DEF(IPHASE,I)=U(I)
     DO 34 I=1,NDF
     IF(ABS(U(I)).GE.1.5)PRINT 93
93   FORMAT(1H ,///,40X,' SECTION SIZES INADEQUATE',/,
    #41X,24('*'))
     IF(ABS(U(I)).GE.1.5) STOP
     IF(INDEX.GE.5) PRINT 187
187  FORMAT(1H ,' SOLUTION NOT OBTAINED AFTER FIVE ITERATIONS')
     IF(INDEX.GE.5) STOP
34   CONTINUE
     RETURN
     END
```

```
PLASLM.ADPF(1).ITED
                         COMPILER (XM=3)
                         SUBROUTINE ITED
                         COMMON /EXT/KK(157,98)
               C         ************************************************
               C         ITERATION PROCEDURE IN WHICH SIGNUM FUNCTION
               C         IS ASSIGNED A VALUE ACCORDING TO THE SIGN OF
               C         THE BENDING MOMENT AT A POINT IN THE STRUCTURE
               C         UNDERGOING OR ABOUT TO UNDERGO PLASTIC DEFORMATION
               C         ************************************************
                         COMMON/BLK1/NROWP,ICOLP,MP(23,2),NE,NDF,U(73),P(73),
                        #NE,IP,LTS(23,4,2),PSTRN(23,4),STRN(23,4),STRS(23,4),
                        #NBWA,ISIG(23,4),NELTS,LP,NB,M,ISTOP,MAX,
                        #NBW,NJ,E,EP,AE,EI,YSTRS,NBEAM(23,2),COORDX(23),
                        #COORDY(23),ELLEN(23),SELT(23),CELT(23),NDFA,NDFS,
                        #NBWS,NDFBWA,IBC(73),N1,N2,IIBC(3),PP(10,73),RLAMDA(10),
                        #PPP(3),WL(3),WLOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
                        #B(23,4,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALPHA(23,2),
                        #DSTRN(23,4),DPSTRN(23,4),DSTRS(23,4),NELT(88,4),DU(73),
                        #DP(73),RMIN,IDUM,F1(73),MS,NBS,EA(23),RE1(23),IPHASE,
                        #INDEX,IDUM1,LDCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
                        #IEND(23),NLC,NSEC
                         REAL MP
                         DO 1 IE=1,NE
                         DO 1 I=3,4
                         II=I-2
                         IF(STRS(IE,I)/MP(IE,II).GE.0.999.AND.(PSTRN(IE,I)).GE.0.0)
                        * ISIG(IE,I)=1.
                         IF(STRS(IE,I)/MP(IE,II).LE.-0.999.AND.(PSTRN(IE,I)).LE.0.0)
                        * ISIG(IE,I)=-1.
                         IF(ISIG(IE,I).EQ.1.AND.PSTRN(IE,I).LT.0.) PSTRN(IE,I)=0.
                         IF(ISIG(IE,I).EQ.-1.AND.PSTRN(IE,I).GT.0:0)
                        *PSTRN(IE,I)=0.
                    1    CONTINUE
                         INDEX=INDEX+1
                         CALL SOLVED
                         CALL STRSTD
                         RETURN
                         END

@PRT,S ADPF.SORT,.CONTRL,.CHECK,.NEWSEC
```

```
PLASL:I.ADPF(1).SORT
      COMPILER (XM=3)
      SUBROUTINE SORT
      COMMON /EXT/KK(157,98)
C      *************************
C      CHOOSE SUITABLE SECTIONS
C      *************************
      COMMON/BLK1/NROWP,ICOLP,MP(23,2),NF,NDF,U(73),P(73),
     #NE,IPELTS(23,4,2),PSTRN(23,4),STRN(23,4),STRS(23,4),
     #NBWA,ISIG(23,4),NELTS,LP,NB,M,ISTOP,MAX,
     #NBW,NB,E,EP,AE,EI,YSTRS,NBEAM(23,2),COORDX(23),
     #COORDY(23),ELLEN(23),SELT(23),CELT(23),NDFA,NDFS,
     #NBWS,NDFBWA,IBC(73),N1,N2,IIBC(3),PP(10,73),RLAMDA(10),
     #PPP(3),NL(3),WLOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
     #B(23,4,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALPHA(23,2),
     #DSTRN(23,4),DPSTRN(23,4),DSTRS(23,4),NELT(88,4),DU(73),
     #DP(73),RMIN,IDUM,P1(73),MS,NBS,EA(23),REI(23),IPHASE,
     #INDEX,IDUM1,LDCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
     #IEND(23),NLC,NSEC
      COMMON/BLK3/RLDFCT(30),NL1,DLDFCT(30),ALDFCT(30)
      COMMON/BLK6/SSP(23,20)
      COMMON/BLK7/LCRTCS
      COMMON/BLK9/IHP,ISHORT,ITAB
      DIMENSION ZZP(23),ISEC(23)
C      ****************************
C      DETERMINE CRITICAL LOAD CASE
C      ****************************
      IF(IDUM.EQ.100) GO TO 3457
      IDUM=0
      IPHASE=1
      REAL MP
      DUM=1.E30
      DO 3 I=1,NLC
      IF(ALDFCT(I).GE.DUM) GO TO 3
      DUM=ALDFCT(I)
      LCRTCS=I
    3 CONTINUE
      IK=1
      IJ=1
C     ********************************
C     DETERMINE YIELD MOMENTS REQUIRED
C     ********************************
      DO 2 IE=1,NE
      MP(IE,1)=MP(IE,1)*100./ALDFCT(LCRTCS)
    2 CONTINUE
      IF(ISHORT.EQ.1) GO TO 10
      PRINT 9
    9 FORMAT(1H ,///,11X,' YIELD MOMENTS REQUIRED',/,
     #40X,' ELEMENT')
      PRINT 4, (IE,MP(IE,1),IE=1,NE)
    4 FORMAT(1H ,41X,I3,12X,E11.6,/)
   10 REWIND 12
      READ(12,100) ((SP(I,J),J=1,20),I=1,NSEC)
  100 FORMAT( )
 3457 PRINT 3456
 3456 FORMAT(1H1,//,49X,'0',/,40X,'<',18('-'),'>',
     #//40X,20('*'),4X,'A',/,40X,'*',16X,'*',4X,'!',/,
     #40X,'*',13X,'*',4X,'!',/,40X,9('*'),2X,9('*'),4X,
     #'!',6(/,48X,'*',2X,'*',12X,'!'),/,
     #44X,'--->'  '<---T1',6X,'!',1X,'H',20X,'SECTION PROPERTIES',
```

```
      IF(IBP.EQ.1) LDCASE=IPHASE
      IF(IBP.EQ.1) GO TO 99
      IF(LDDUM.EQ.LCRTCS) GO TO 400
      IF(IPHASE.GT.1) PRINT 240, LDDUM
  240 FORMAT(1H1,'  LOAD CASE',1X,I2,1X,' MUST NOW BE CHECKED')
      IF(IPHASE.GT.1) GO TO 99
  999 CALL FLTMAD
      IF(RMIN.EQ.1) PRINT 3
    3 FORMAT(1H ,////,16X,' HAVING OBTAINED NEW SECTIONS THE'
     #' STRUCTURE MUST BE REANALYSED USING DEFORMATION THEORY')
   99 DO 77 I=1,NDF
   77 U(I)=0
      DO 88 IE=1,NE
      DO 88 J=1,4
      PSTRN(IE,J)=0
      STRN(IE,J)=0
      STRS(IE,J)=0
      ISIG(IE,J)=0
   88 CONTINUE
      IF(IBP.NE.1.AND.IPHASE.GT.1) LDCASE=LDDUM
      CALL SOLVED
      CALL STRSTD
      RETURN
      END
```

```
PLASLM*ADPF(1).CHECK
      COMPILER(XM=3)
      SUBROUTINE CHECK
      COMMON /EXT/RK(157,98)
C     *************************************
C     CHECKS THE CHOICE OF SECTIONS ACCORDING
C     TO THE A.I.S.C. SPECIFICATIONS
C     *************************************
      COMMON/BLK1/NROWP,NCOLP,MP(23,2),NF,NDF,U(73),P(73),
     #NE,IPELTS(23,4,2),PSTRN(23,4),STRN(23,4),STRS(23,4),
     #NBWA,ISIG(23,4),NELTS,LP,NB,M,ISTOP,MAX,
     #NBW,NJ,E,EP,AE,EI,YSTRS,NBEAM(23,2),COORDX(23),
     #COORDY(23),ELLEN(23),SELT(23),CELT(23),NDFA,NDFS,
     #NBWS,NDFBWA,IBC(73),N1,N2,IIBC(3),PP(10,73),RLAMDA(10),
     #PPP(3),WL(3),WLOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
     #B(23,4,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALPHA(23,2),
     #DSTRN(23,4),DPSTRN(23,4),DSTRS(23,4),NELT(88,4),DU(73),
     #DP(73),RMIN,IDUM,P1(73),MS,JBS,EA(23),REI(23),IPHASE,
     #INDEX,IDUM1,LDCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
     #IEND(23),NLC,NSEC
      COMMON/BLK2/CSTRS(10,23,4),EFFLEN(23),ICOL(10),
     #CM1(23),CM(23),PE(23),BUKCHK(23,2),BM(23),RMM(23),
     #DMP(23),RETURN(10,23)
      COMMON/BLK5/RLBB(10,23),SRATIO(10,23),AXSTR(10,23)
      COMMON/BLK6/SSP(23,20)
      COMMON/BLK7/LCRTCS
      COMMON/BLK9/IBP,ISHORT,ITAB
      COMMON/BLK10/EFFMOM(23)
      DIMENSION PCR(23),IT(7),IBMCOL(23)
      DATA IAP,IBBP,ICP/'N/A','NO','OK'/
      REAL MP
      ISSTOP=0
      LDCASE=LCRTCS
      IF(IBP.EQ.1) LDCASE=1
  241 I=I+1
      IF(ABS(SSP(I,17)-SSP(I+1,17)).LE.1.E-8) IFLAG=-1
      IF(IFLAG.EQ.-1) GO TO 239
      IF(I+1.LT.NE) GO TO 241
  239 CONTINUE
      PRINT 1
    1 FORMAT(1H1,54X,22('*'),/,54X,' COMMENCE DESIGN CHECKS',/,
     #55X,22('*'),/,43X,'( IN ACCORDANCE WITH A.I.S.C.'
     #' SPECIFICATIONS )',////)
      IF(IBP.EQ.1) PRINT 93,LDCASE
   93 FORMAT(1H ,//,1X,13('*'),/,' LOAD CASE',1X,I2,/,1X,13('*'))
C
C     CHECK B/T2 RATIOS : FLANGE/WEB BUCKLING
C
      DO 1000 IE=1,NE
      IF(ITAB.EQ.1.AND.IE.EQ.1.AND.IBP.NE.1) PRINT 6785
 6785 FORMAT(1H ,47X,'LOCAL BUCKLING',/,1H ,'ELEMENT'
     #,2X,'FLANGES(B/T RATIO)',1X,'WEBS(H/T1 RATIO)',
     #1X,'SHEAR,WEB CRIPPLING',1X,'EFF.YIELD MOMENT',
     #1X,'SLENDERNESS RATIO',1X,'AXIAL STRNTH',1X,
     #'IN PLANE BUCKLING',/,1H ,9X,'/',16('-'),'/',1X,
     #'/',14('-'),'/',1X,'/',17('-'),'/',1X,'/',14('-'),'/',
     #1X,'/',15('-'),'/',1X,'/',10('-'),'/',1X,'/',15('-'),'/')
      IBMCOL(IE)=1
      IF(ICOL(IE).EQ.IE) IBMCOL(IE)=2
      RETURN(LDCASE,IE)=0
```

```
1000   CONTINUE
94     IF(IBP.EQ.1) PRINT 6785
       DO 2 IE=1,NE
       IPP=0
       DO 342 II=1,7
342    IT(II)=0
5      FORMAT(1H ,///,' INCREASE SECTION FOR ELEMENT',I3)
       IF(IFLAG.NE.-1) PRINT 8,IE
8      FORMAT(1H ,///,11('*'),/,' ELEMENT',I3,/,11('*'))
       IF(IBMCOL(IE).EQ.1.AND.ISHORT.NE.1) PRINT 24,IE
       IF(IBMCOL(IE).EQ.2.AND.ISHORT.NE.1) PRINT 25,IE
24     FORMAT(1H ,' ELEMENT',I3,'  IS A BEAM')
25     FORMAT(1H ,' ELEMENT',I3,'  IS A COLUMN')
       IF((SSP(IE,5)/SSP(IE,7)).LE.17.AND.ISHORT.NE.1) PRINT 3
       IF((SSP(IE,5)/SSP(IE,7)).LE.17.AND.ISHORT.EQ.1
      #.AND.ITAB.EQ.1) IT(1)=1
3      FORMAT(1H ,//,' B/T RATIO OK')
       IF((SSP(IE,5)/SSP(IE,7)).GT.17) PRINT 4,IE
       IF((SSP(IE,5)/SSP(IE,7)).GT.17) STOP
4      FORMAT(1H ,//,' ELEMENT',I3,' IS NOT COMPACT'
      #' AND MAY NOT BE USED FOR PLASTIC DESIGN')
C
C      CHECK H/T1 RATIO : WEBS
C
       RATIO1=ABS(CSTPS(LDCASE,IE,1))/(YSTRS*SSP(IE,10))
       SUM1=1100*(1-1.44*RATIO1)/SQRT(YSTRS*1.E-6)
       SUM2=680/SQRT(YSTRS*1.E-6)
       RATIO2=(SSP(IE,4)/SSP(IE,6))
       IF(RATIO1.LE.0.265.AND.SUM1.GE.RATIO2.AND.ISHORT.NE.1) PRINT 6
       IF(RATIO1.GT.0.265.AND.SUM2.GE.RATIO2.AND.ISHORT.NE.1) PRINT 6
       IF(RATIO1.LE.0.265.AND.SUM1.GE.RATIO2.AND.ISHORT.EQ.1
      #.AND.ITAB.EQ.1) IT(2)=1
       IF(RATIO1.GT.0.265.AND.SUM2.GE.RATIO2.AND.ISHORT.EQ.1
      #.AND.ITAB.EQ.1) IT(2)=1
       IF(RATIO1.LE.0.265.AND.SUM1.LT.RATIO2) PRINT 7
       IF(RATIO1.GT.0.265.AND.SUM2.LT.RATIO2) PRINT 7
       IF(RATIO1.LE.0.265.AND.SUM1.LT.RATIO2) IPP=1
       IF(RATIO1.GT.0.265.AND.SUM2.LT.RATIO2) IPP=1
       IF(IPP.EQ.1) PRINT 4,IE
       IF(IPP.EQ.1) STOP
6      FORMAT(1H ,//,' H/T1 RATIO FOR SPECIFIED N/NP RATIO OK')
7      FORMAT(1H ,//,' H/T1 RATIO FOR SPECIFIED N/NP RATIO UNACCEPTABLE')
C
C      SHEAR AND WEB CRIPPLING
C
       I=0
9325   I=I+1
       IF(I.GT.2) GO TO 9324
       DUM=0.55*YSTRS*SSP(IE,4)*SSP(IE,6)*1.E-9
       IF(ABS(CSTRS(LDCASE,IE,2)).LE.DUM.AND.ISHORT.NE.1) PRINT 9
       IF(ABS(CSTRS(LDCASE,IE,2)).LE.DUM.AND.ISHORT.EQ.1
      #.AND.ITAB.EQ.1) IT(3)=1
       IF(ABS(CSTRS(LDCASE,IE,2)).GT.DUM.AND.ISHORT.NE.1) PRINT 99
       IF(ABS(CSTRS(LDCASE,IE,2)).GT.DUM) IPP=1
       IF(ISHORT.NE.1) PRINT 66,CSTRS(LDCASE,IE,2),DUM
66     FORMAT(1H ,' SHEAR FORCE APPLIED',5X,' SHEAR RESISTANCE'
      #,/,5X,E10.5,15X,E10.5)
       IF(IPP.EQ.1) PRINT 77
77     FORMAT(1H ,' DIAGONAL STIFFENERS ARE REQUIRED')
9      FORMAT(1H ,//,' SHEAR AND WEB CRIPPLING CHECK OK')
```

```
 99      FORMAT(1H ,//,' SHEAR AND WEB CRIPPLING CHECK UNACCEPTABLE')
         DUM=0.55*YSTRS*SSP(IE,4)*SSP(IE,6)*1.E-9
         DUM1=1.E-9*SSP(IE,6)*((SSP(IE,4)/2)-SSP(IE,7))**2.
         DUM2=1-SQRT(1-(CSTRS(LDCASE,IE,2)/DUM)**2)
         DUM3=(YSTRS*SSP(IE,13)*1.E-7)*(1-(DUM1/(SSP(IE,13)*1.E-6))
        #*DUM2)
         A1=ABS(CSTRS(LDCASE,IE,3))
         A2=ABS(CSTRS(LDCASE,IE,4))
         A=AMAX0(A1,A2)
         AA=DUM3/A
         IF(AA.GE.0.98.AND.ISHORT.EQ.1.AND.ITAB.EQ.1) IT(4)=1
         IF(AA.GE.0.98.AND.ISHORT.NE.1) PRINT 100
         IF(AA.LT.0.98) IPP=1
         IF(ISHORT.NE.1) PRINT 889,DUM3,CSTRS(LDCASE,IE,I+2)
 889     FORMAT(1H ,' EFFECTIVE MOMENT SUPPLIED',11X,
        #' MOMENT APPLIED',/,/X,E10.5,24X,E10.5)
         IF(IPP.EQ.1) RETURN(LDCASE,IE)=1
         IF(IPP.EQ.1) EFFMOM(IE)=DUM3
         IF(IPP.EQ.1) PRINT 1111,IE
 1111    FORMAT(1H ,'EFFECTIVE YIELD MOMENT FOR ELEMENT '
        #,I2,' UNACCEPTABLE')
 100     FORMAT(1H ,//,' EFFECTIVE YIELD MOMENT OK ')
         GO TO 9325
 9324    CONTINUE
 C
 C       COLUMNS AND BEAM-COLUMNS
 C
 C       COLUMNS
 C
         IF(IBMCOL(IE).EQ.1.AND.ITAB.EQ.1) IT(5)=-1
         A1=ELLEN(IE)/(SSP(IE,18)*1.E-3)
         A2=EFFLEN(IE)*ELLEN(IE)/(SSP(IE,14)*1.E-3)
         IF(A1.GT.A2) SLNDER=A1
         IF(A2.GE.A1) SLNDER=A2
         CC=SQRT(19.739209*E/YSTRS)
         A1=YSTRS*1.E-3*(1-((SLNDER*SLNDER)/(2*CC*CC)))
         A2=1.6666667+(0.375*SLNDER/CC)-((SLNDER/CC)**3)/8
         PCR(IE)=1.7*SSP(IE,10)*1.E-3*A1/A2
         IF(IBMCOL(IE).EQ.1) GO TO 13
         IF(CC.GT.SLNDER.AND.ISHORT.NE.1) PRINT 14
         IF(CC.GT.SLNDER.AND.ISHORT.EQ.1) IT(5)=1
         IF(CC.LE.SLNDER) PRINT 15,IE
         IF(ISHORT.NE.1) PRINT 333,CC,SLNDER
 333     FORMAT(1H ,4X,' CC',9X,' EFFECTIVE SLENDERNESS RATIO'
        #,/,E10.5,14X,E10.5)
         IF(CC.LE.SLNDER) RETURN(LDCASE,IE)=1
         IF(RETURN(LDCASE,IE).EQ.1) SRATIO(LDCASE,IE)=SLNDER/CC
         IF(RETURN(LDCASE,IE).EQ.1.AND.ITAB.NE.1)PRINT 5,IE
         IF(RETURN(LDCASE,IE).EQ.1) GO TO 2
 14      FORMAT(1H ,//,' SLENDERNESS RATIO OK')
 15      FORMAT(1H ,//,10X,' SLENDERNESS RATIO UNACCEPTABLE'
        #' FOR ELEMENT',1X,I2)
         IF(PCR(IE).GE.ABS(CSTRS(LDCASE,IE,1))
        #.AND.ISHORT.NE.1) PRINT 16
         IF(PCR(IE).LT.ABS(CSTRS(LDCASE,IE,1)).AND.ITAB.NE.1) PRINT 17,IE
         IF(PCR(IE).LT.ABS(CSTRS(LDCASE,IE,1))) RETURN(LDCASE,IE)=1
         IF(RETURN(LDCASE,IE).EQ.1) PRINT 5,IE
         IF(ISHORT.NE.1) PRINT 444,PCR(IE),CSTRS(LDCASE,IE,1)
 444     FORMAT(1H ,'MAXIMUM AXIAL STRENTH',5X,' APPLIED'
        #' AXIAL LOAD',/,5X,E10.5,15X,E10.5)
```

```
          IF(PCR(IE).GE.ABS(CSTRS(LDCASE,IE,1)).AND.ISHORT
         #.EQ.1.AND.ITAB.EQ.1) IT(6)=1
   13     IF(ISHORT.EQ.1.AND.ITAB.EQ.1.AND.IBMCOL(IE).EQ.1) IT(6)=-1
          IF(RETURN(LDCASE,IE).EQ.1) AXSTR(LDCASE,IE)=
         #ABS(CSTRS(LDCASE,IE,1))/PCR(IE)
          IF(RETURN(LDCASE,IE).EQ.1) GO TO 2
   16     FORMAT(1H ,//,' MAXIMUM AXIAL STRENTH OK')
   17     FORMAT(1H ,//,' MAXIMUM AXIAL STRENTH UNACCEPTABLE')
   C
   C      BEAM-COLUMNS
   C
          BM(IE)=CSTRS(LDCASE,IE,4)
          IF(CSTRS(LDCASE,IE,3).GE.CSTRS(LDCASE,IE,4))
         #BM(IE)=CSTRS(LDCASE,IE,3)
          DMP(IE)=MP(IE,2)
          IF(MP(IE,1).GE.MP(IE,2)) DMP(IE)=MP(IE,1)
          DUM1=SSP(IE,18)*1.E-3
          RMM(IE)=(1.07-((ELLEN(IE)*SQRT(YSTRS))/
         #(8300*DUM1)))*DMP(IE)
          IF(CM1(IE).EQ.4)  CM(IE)=1.0
          IF(CM1(IE).EQ.3) CM(IE)=0.85
          DUM=(CSTRS(LDCASE,IE,3)/CSTRS(LDCASE,IE,4))
          IF(DUM.GE.1) DUM=1/DUM
          IF(CM1(IE).EQ.2) CM(IE)=0.6-0.4*DUM
          IF(CM1(IE).EQ.2.AND.CM(IE).LT.0.4) CM(IE)=0.4
          IF(CM1(IE).EQ.1) CM(IE)=0.85
          DUM1=SSP(IE,14)*1.E-3
          PE(IE)=1.E-3*E*((3.1415926*DUM1)/(ELLEN(IE)*EFFLEN(IE)))**2
          IF(ISHORT.NE.1) PRINT 22
   22     FORMAT(1H ,//,' INTERACTION FORMULA ONE',/,24('*'),/)
          DUM1=CSTRS(LDCASE,IE,1)/PCR(IE)
          DUM2=(CM(IE)*BM(IE))/RMM(IE)
          DUM3=(1-(CSTRS(LDCASE,IE,1)/PE(IE)))
          BUKCHK(IE,1)=DUM1+DUM2/DUM3
          IF(BUKCHK(IE,1).GT.1) PRINT 20
          IF(BUKCHK(IE,1).GT.1) ISSTOP=1
          IF(RETURN(LDCASE,IE).EQ.1) PRINT 5,IE
          IF(ISHORT.NE.1) PRINT 555,BUKCHK(IE,1)
   555    FORMAT(1H ,' FORMULA ONE',5X,' REQUIRED : LESS THAN'
         #' OR FQUAL TO ONE',/,3X,E10.5)
   20     FORMAT(1H ,//,' INTERACTION FORMULA ONE UNACCEPTABLE')
   21     FORMAT(1H ,//,' INTERACTION FORMULAE ACCEPTABLE')
   2000   FORMAT(1H ,//,' INTERACTION FORMULA TWO UNACCEPTABLE')
          IF(RETURN(LDCASE,IE).EQ.1) GO TO 2
          DUM1=CSTRS(LDCASE,IE,1)/(YSTRS*SSP(IE,10)*1.E-6)
          DUM2=BM(IE)/(1.18*DMP(IE))
          BUKCHK(IE,2)=DUM1+DUM2
          IF(ISHORT.NE.1) PRINT 23
   23     FORMAT(1H ,//,' INTERACTION FORMULA TWO',/,24('*'),/)
          IF(ISHORT.NE.1) PRINT 666,BUKCHK(IE,2)
   666    FORMAT(1H ,' FORMULA TWO',5X,' REQUIRED : LESS THAN'
         #' OR FQUAL TO ONE',/,3X,L10.5)
          IF(BUKCHK(IE,2).GT.1) PRINT 2000
          IF(BUKCHK(IE,1).LE.1.AND.BUKCHK(IE,2).LE.1
         #.AND.ISHORT.NE.1) PRINT 21
          IF(BUKCHK(IE,1).LE.1.AND.BUKCHK(IE,2).LE.1
         #.AND.ISHORT.EQ.1.AND.ITAB.EQ.1) IT(7)=1
          IF(BUKCHK(IE,2).GT.1) ISSTOP=1
          IF(ISSTOP.EQ.1) PRINT 5,IE
          IF(ISSTOP.EQ.1) GO TO 2
```

```
      DO 123 I=1,7
      IF(IT(I)) 111,112,113
111   IT(I)=IAP
      GO TO 123
112   IT(I)=IBBP
      GO TO 123
113   IT(I)=ICP
123   CONTINUE
      PRINT 103,IE,(IT(I),I=1,7)
103   FORMAT(1H0,2X,I2,2X,':',8X,A2,16X,A2,19X,A2,17X,A2,
     #16X,A3,12X,A3,13X,A2,/)
2     CONTINUE
      IF(ISSTOP.EQ.1) STOP
C
C     CHECK LATERAL BRACING OF STRUCTURE
C
      DO 2356 IE=1,NE
      IF(RETURN(LDCASE,IE).EQ.1) IPPQ=1
2356  CONTINUE
      IF(ITAB.EQ.1.AND.IPPQ.NE.1) PRINT 1322
      DO 222 IE=1,NE
1322  FORMAT(1H ,///,50X,15('*'),/,50X,
     #'LATERAL BRACING',/,50X,15('*'))
      IF(ISHORT.NE.1.AND.IPPQ.NE.1) PRINT 1322
      IDUM11=0
      DO 223 IJ=1,2
      IIDUM=0
      A=ABS(CSTRS(LDCASE,IE,IJ+2))
      IF((MP(IE,IJ)-A).GE.1.) GO TO 33
      IF((MP(IE,IJ)-A).LT.1) IIDUM=1
      IF((MP(IE,IJ)-A).LT.1.AND.IPPQ.NE.1)PRINT 31,NBEAM(IE,IJ),IE
31    FORMAT(1H ,///,' PLASTIC HINGE AT NODE',1X,I2,' OF ELEMENT',1X,I2)
      I=1
      DUM1=(SSP(IE,18)*1.E-3)*((9500./(YSTRS*1.E-6))+25.)
      DUM2=(SSP(IE,18)*1.E-3)*((9500./(YSTRS*1.E-6)))
      IF(IPPQ.NE.1) PRINT 1312,DUM1,DUM2
1312  FORMAT(1H ,30X,'FOR :',5X,'SINGLE CURVATURE',14X,
     #'DOUBLE CURVATURE',/,27X,'CRITICAL',8X,E10.5,20X,
     #E10.5,/,28X,'LENGTH',/)
      IF(IIDUM.EQ.1) GO TO 223
33    IF(IDUM11.EQ.1) GO TO 223
      IDUM11=1.
      DUM=SQRT(YSTRS*1.E-6)
      DUM1=SSP(IE,5)*0.23028/DUM
      DUM2=1.68737E2*SSP(IE,5)*SSP(IE,7)/(YSTRS*SSP(IE,4)*1.E-6)
      A=MIN(DUM1,DUM2)
      IF(IFLAG.NE.-1.AND.IPPQ.NE.1) PRINT 1332,A
1332  FORMAT(1H ,' LIMITING UNBRACED LENGTH :',1X,E10.5)
      IF(IFLAG.EQ.-1.AND.IE.EQ.1.AND.IPPQ.NE.1) PRINT 1333,A
1333  FORMAT(1H ,///,' LIMITING UNBRACED LENGTH FOR'
     #' NON-PLASTIC ELEMENTS :',2X,E10.5)
223   CONTINUE
222   CONTINUE
      IF(IBP.NE.1) GO TO 778
      IF(LDCASE.EQ.NLC) GO TO 778
      LDCASE=LDCASE+1
      PRINT 985,LDCASE
985   FORMAT(1H1,12('*'),/,' LOAD CASE',1X,I2,/,13('*'))
      GO TO 94
778   LDCASE=0
```

```
  777  IE=0
       LDCASE=LDCASE+1
 6666  IE=IE+1
       IF(IBP.NE.1) LDCASE=LCRTCS
       IF(IBP.NE.1) NLC=LDCASE
       IF(LDCASE.EQ.NLC.AND.IE.EQ.NE) GO TO 395
       IF(RETURN(LDCASE,IE).EQ.1) CALL NEWSEC
       IF(IE.EQ.NE) GO TO 777
       GO TO 6666
  395  PRINT 2346
 2346  FORMAT(1H ,///,39X,' ALL DESIGN CHECKS'
      #' SUCCESSFULLY COMPLETED')
       PRINT 3658
 3658  FORMAT(1H1,' PHEW')
       STOP
       END
```

```
PLASLM.ADPF(1),NEWSEC
        COMPILER(XM=3)
        SUBROUTINE NEWSEC
        COMMON /EXT/RK(157,98)
C       ********************************************
C       IF THE A.I.S.C. REQUIREMENTS ARE NOT MET
C       A NEW SET OF SECTIONS HAS TO BE CHOSEN
C       ********************************************
        COMMON/BLK1/NROWP,NCOLP,MP(23,2),NF,NDF,U(73),P(73),
       #NE,IPFLTS(23,4,2),PSTRN(23,4),STRN(23,4),STRS(23,4),
       #NBWA,ISIG(23,4),NELTS,LP,NB,M,ISTOP,MAX,
       #NBW,NN,E,EP,AE,EI,YSTRS,NBEAM(23,2),COORDX(23),
       #COORDY(23),ELLEN(23),SELT(23),CELT(23),NDFA,NDFS,
       #NBWS,NDFBWA,IBC(73),N1,N2,I1BC(3),PP(10,73),RLAMDA(10),
       #PPP(3),NL(3),VLOAD(10,73),OUT(73),DET,FACT,S(23,4,4),
       #B(23,4,6),BTS(23,6,4),BTSB(6,6),DEL(6),ALPHA(23,2),
       #DSTRN(23,4),DPSTRN(23,4),DSTRS(23,4),NELT(88,4),DU(73),
       #DP(73),RMIN,IDUM,P1(73),MS,NBS,EA(23),REI(23),IPHASE,
       #INDEX,IDUM1,LDCASE,SP(42,20),ZP(23,2),ICOUNT(23,2),
       #IEND(23),NLC,NSEC
        COMMON/BLK2/CSTRS(10,23,4),EFFLEN(23),ICOL(10),
       #CM1(23),CM(23),PE(23),BUKCHK(23,2),BM(23),RMM(23),
       #DMP(23),RETURN(10,23)
        COMMON/BLK5/RLBB(10,23),SRATIO(10,23),AXSTR(10,23)
        COMMON/BLK6/SSP(23,20)
        COMMON/BLK7/LCRTCS
        COMMON/BLK9/IBP,ISHORT,ITAB
        COMMON/BLK10/EFFMOM(23)
        DIMENSION SSP1(23,20),SSP2(23,20),ICNT(23)
       #,ICNT1(23),ISEC(23),RLBBD(23),SRTIOD(23),AXSTRD(23)
        REAL MP
        REWIND 12
        READ(12,100)((SP(I,J),J=1,20),I=1,NSEC)
  100   FORMAT( )
        PRINT 111
  111   FORMAT(1H1,/,55X,12('*'),/,55X,'NEW SECTIONS',/,
       #55X,12('*'))
        PRINT 666
  666   FORMAT(1H ,///,' ELEMENT',7X,'SECTION',5X,
       #'H',7X,'B',7X,'T1',6X,'T2',6X,'A',6X,'IXX',6X,'ZEXX'
       #,4X,'ZPXX',4X,'RXX',5X,'IYY',5X,'ZEYY',4X,
       #'ZPYY',4X,'RYY')
        IF(IBP.NE.1) GO TO 14
        I1=0
   12   I2=0
        I1=I1+1
   13   I2=I2+1
        IF(RETURN(LDCASE,IE).NE.1) GO TO 22
        IF(I1.EQ.NLC.AND.I2.EQ.NE) GO TO 14
        IF(RLBB(I1,I2).LT.1.E-8) GO TO 20
        IF(RLBB(I1,I2).GE.RLBB(I1+1,I2)) RLBBD(I2)=RLBB(I1,I2)
   20   IF(AXSTR(I1,I2).LT.1.E-8) GO TO 21
        IF(AXSTR(I1,I2).GE.AXSTR(I1+1,I2)) AXSTRD(I2)=AXSTR(I1,I2)
      21 IF(SRATIO(I1,I2).LT.1.E-8) GO TO 22
        IF(SRATIO(I1,I2).GE.SRATIO(I1+1,I2)) SRTIOD(I2)=SRATIO(I1,I2)
   22   IF(I2.EQ.NE) GO TO 12
        GO TO 13
   14   CONTINUE
        IF(IBP.EQ.1) GO TO 144
        DO 933 IE=1,NE
```

```
          RLBBD(IE)=RLBB(LDCASE,IE)
          SRTIOD(IE)=SRATIO(LDCASE,IE)
          AXSTRD(IE)=AXSTR(LDCASE,IE)
  933     CONTINUE
  144     DO 1 IE=1,NE
          IF(RETURN(LDCASE,IE).NE.1) GO TO 1
          IK=0
  331     IK=IK+1
          IF((SSP(IE,13)-SP(IK,13)).GE.0) GO TO 331
          ISTART=IK
          IF(ABS(RLBBD(IE)).GT.1) SSP1(IE,18)=SSP(IE,18)
         #*ABS(RLBBD(IE))
          IF(ABS(SRTIOD(IE)).GT.1) SSP2(IE,18)=SSP(IE,18)
         #*ABS(SRTIOD(IE))
          IF(ABS(AXSTRD(IE)).GT.1) SSP(IE,10)=SSP(IE,10)*ABS(AXSTRD(IE))
          SSP(IE,18)=SSP2(IE,13)
          IF(SSP1(IE,18).GE.SSP2(IE,13)) SSP(IE,18)=SSP1(IE,18)
          IK=ISTART
  3       IK=IK+1
          IF(IK.GT.NSEC) ISEC(IE)=1
          IF((SSP(IE,18)-SP(IK,18)).GT.0) GO TO 3
          ICNT(IE)=IK
          IK=ISTART
  5       IK=IK+1
          IF(IK.GT.NSEC) ISEC(IE)=1
          IF((SSP(IE,10)-SP(IK,10)).GT.0) GO TO 5
          ICNT1(IE)=IK
          II=ICNT(IE)
          IF(ICNT1(IE).GE.ICNT(IE)) II=ICNT1(IE)
          IK=0
  6       IK=IK+1
          IF(IK.GT.NSEC) ISEC(IE)=1
          IF((SSP(IE,10)-SP(IK,10)).GT.0) GO TO 6
          ICNT(IE)=IK
          IJK=II
          IF(ICNT(IE).GT.II) IJK=ICNT(IE)
  400     CONTINUE
          DO 7 IJ=1,20
  7       SSP(IE,IJ)=SP(IJK,IJ)
          EA(IE)=SSP(IE,10)*E*1.E-6
          REI(IE)=SSP(IE,11)*E*1.E-9
          MP(IE,1)=YSTRS*SSP(IE,13)*1.E-9
          IF(EFFMOM(IE).LT.1.) GO TO 3221
          IF((MP(IE,1)/EFFMOM(IE)).LT.0.98) MP(IE,1)=EFFMOM(IE)
   3221   MP(IE,2)=MP(IE,1)
  545     FORMAT(1H ,' NO SUITABLE SECTION IS AVAILABLE!
         #' FROM SELECTION IN DATA-FILE :  FOR ELEMENT',1X,I2)
   1      CONTINUE
          DO 55 IE=1,NE
          IF(ISEC(IE).EQ.1) PRINT 545,IE
          PRINT 777,(IE,SSP(IE,1),SSP(IE,2),(SSP(IE,J),J=4,7),
         #(SSP(IE,J1),J1=10,18))
  777     FORMAT(1H ,2X,I3,8X,F4.0,'*',F4.0,2X,5(F5.1,3X)
         #,3(F6.1,2X),5(F5.1,3X),/)
   55      CONTINUE
          DO 555 IE=1,NE
   555    IF(ISEC(IE).EQ.1) STOP
          INDEX=1
          IPHASE=1
          RMIN=1
```

```
     120                  IF(ISP.EQ.1) LOCASE=1
     121                  IF(ISP.EQ.1) LP=1
     122                  LOCASE=LCRTCS
     123                  LP=LCRTCS
     124                  CALL CONTRL
     125                  END

  JOBKPT PRINTS
```

```
GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF
GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF
GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF
GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF GRIF
12345678901234567890123456789012345678901234567890123456789012345678901234567890
```