A software package for the structural analysis of large plated structures with particular reference to independent prismatic tanks for the carriage of liquefied petroleum gas by ship.

by

Charles Frederick Smith

Thesis submitted for the degree of

Doctor of Philosophy

The City University, London.

Department of Aeronautics.

July 1984

Contents. Page

Appendices.

Subroutine Names.

Function Names.

List of Figures.

Test 1.1 Idealization of beam.

Test 1.2 Accuracy of four node compatible rectangular elements plotted against the idealization's number of degrees of freedom for a plate beam.

Test 1.3 Tabulation of accuracy figures for isoparametric and distortion mode elements.

Test 1.4 Plot showing percentages of time saved, achieved by using distortion mode elements as opposed to low order compatible elements.

Test 1.5 Quarter beam represented by the data in Fig's Test 1.6, Test 1.7 and Test 1.8.

Test 1.6 Data to run present package for the beam shown in Fig. Test 1.5.

Test 1.7 As Test 1.6 for the Lusas package.

Test 1.8 As Test 1.6 for the Sap IV package.

Test 2.1 Idealization of two cantilever beams connected by compression only springs.

Test 2.2 Plot of stresses found from the current package and EBT for the beam and spring arrangement shown in Fig. Test 2.1.

Test 2.3 Alternative arrangement of Test 2, showing replacement of restraints by springs.

Test 2.4 Comparison of centroid stresses obtained when using fixed restraints (first arrangement) and spring restraints.

Test 2.5 Listing of data used by the present package to analyse the two beams shown in Fig. Test 2.1

Test 2.6 As Test 2.5 for Fig. Test 2.3 arrangement.

Test 3.1 Arrangement of the steel bulkhead used in

the symmetrical symmetrical condition.

Test 9.13 Boundary nodal longitudinal loadings for the anti-symmetrical symmetrical condition.

Test 9.14 Restraints applied to a quarter structure so that the results of the various runs may be summed to give the results for the whole structure.

Test 9.15 Numbering of insulating supports, indicating the order in which the supports were deleted and those which were finally retained.

Test 9.16 Inner bottom and gas tank profiles in way of insulating support blocks. No allowance made for temperature variations.

Test 9.17 Inner bottom and gas tank profiles in way of support blocks when the structure is subjected to all the loads of Test 9. N.B. Fig. Test 9.16 is similarly loaded but does not have the temperature variations given in Table Fig. Test 9.9.

Test 9.18 Ship's deck, bulkhead and outer bottom deflections at $\frac{1}{4}$ beam, whilst the ship is subjected to all the loads of Test 9.

Test 9.19 Predicted longitudinal section through gas tank at $\frac{1}{4}$ beam off C.L. during test 9.

Test 9.20 Ship's side profile at section 2500m.m fore and aft tank midlength.

Test 9.21 Profile of gas tank sides (knuckle to knuckle) with and without temperature variations.

Test 9.22 Deck longitudinal stresses.

Test 9.23 Outer bottom and lower wing tank longitudinal stresses for test 9.

Test 9.24 Inner and outer bottom longitudinal stresses in way of the row of support blocks nearest the midlength of the gas tank.

Test 9.25 Transverse stresses round the longitudinal centre line plane of the gas tank.

Test 9.26 Distribution of longitudinal stresses at transverse midlength section of gas tank.

Acknowledgements.

In the pursuit of this research and the production of this report, I have received help from many. I am especially grateful to [redacted] [redacted] for her faithful support and forbearance and [redacted] for her typing. I would like to thank Dr. R.F.Williams who provided the initial guidance for this work and always managed to suggest the correct path through various difficulties and Dr. M.D.Wright who ably took over the supervision of this work for its final stages. Finally I am thankful to the Department of Trade for providing some of the time and support necessary to bring this project to a successful conclusion.

Consultation and Copying.

I grant powers of discretion to the University Librarian to allow this thesis to be copied in whole or in part without further reference to me. This permission covers only single copies made for study purposes, subject to normal conditions of acknowledgement.

Abstract.

This work relates to the finite element analysis of three dimensional stiffened plate structures and introduces the idea of a self contained stiffness vector. Such a vector is assembled for each degree of freedom in turn rather than an overlapping stiffness matrix for each element. Using this vector concept and a continuous partial inversion routine, the intermediate version of the stiffness matrix never occupies a space larger than its final size when it is then expressed only in terms of those degrees of freedom which are either automatically or manually nominated to remain active. As a result there is a saving in core space over conventional assembly and wavefront techniques. By this method the wavefront is minimized and goes where it is required at the best time to suit the efficient running of the program. For faster running the offloading of coefficients to disc is delayed until all available core space is exhausted, whereupon a selective clearing of core space is initiated. Since rows of the stiffness matrix corresponding to inactive degrees of freedom are eliminated and finally reformed in their natural order there is a need for only limited housekeeping routines.

This process of a continuous partial inversion shows quite remarkable savings in the time taken for inversion in the analysis of large structures having a few degrees of freedom whose stiffness terms may change or vanish depending upon their deflections during the solution stage and thus require their inversion phase to be within an iterative loop.

The plate contribution to the stiffness matrix employs a little used element whose proof was not available for this work. A derivation of its formulation based on the modes of inplane distortion to which a rectangular plate can be subjected is made and included in the text.

Advantage is taken of the repetitive nature of details in many large structures to take the minimization of hand prepared input data to an extreme. At the same time facilities are available to input information for awkward structures not lending themselves to such a routine. A comprehensive list of error detecting checks monitors each stage of the package. Hydrostatic, thermal and gravity loads are generated by built in routines as are those caused by accelerations of the structure and cargo. Automatic evaluation of stiffness terms is made to prevent singularities associated with floating bodies.

Principal variable names and abbreviations.

AL The breadth of the column of plate elements to the left of the current node.

ALCHEK The breadth of the column of plate elements to the right of the current node.

AMU Poisson's ratio.

BL The height of the first row of plate elements below the current node.

BLCHEK The height of the first row of plate elements above the current node.

E Young's modulus.

IBX Bandwidth of the current element.

IDFA The number of degrees of freedom along the perimeter of a panel.

IDIV The number of plate elements between adjacent stiffeners running parallel to the local y axis of the panel.

IDIVY The number of plate elements between adjacent stiffeners running parallel to the local x axis of the panel.

IMAX A value equal to or smaller than the size of the integer array Idata.

IR The maximum number of degrees of freedom on a panel.

ISX The number of nodes on a panel along the local x axis.

ISXLL The number of nodes on local x axis of a previously processed panel having suitable data for copying by the current panel.

ISY The number of nodes on a panel along the local y axis.

ISYLL To ISY as ISXLL is to ISX.

IT The extent of the array Real available for the storage of the stiffness matrix.

KA The start of temporary storage on the array Idata.

KB The start of the storage of the stiffness matrix.

KBX The total number of degrees of freedom on a panel including restrained freedoms not on the panel's perimeter, spring endings not on the perimeter and freedoms required by an adjacent panel. KBX has at times the same value as NEX.

KE, KI and KL.	Markers are used to indicate the start or finish of blocks of data in the array Idata.
KJ	Number of integers to be put off to disc at the end of FILE.
L	Number of the current panel.
LIMIT	A value equal to or smaller than the real array Real.
LNDF	The local reference number allocated to a degree of freedom.
LOGI	This variable takes a unity value when there is a horizontal stiffener present at the current node, otherwise LOGI has a zero value.
LOGJ	As LOGI but for a vertical stiffener.
M1, M2 M3 and M4.	These variables which are in the common block are used to transfer data between subroutines.
MAXDOF	Is the maximum size of Idata which will be

used before allocating space for the index of stiffness matrix.

MR Disc record number storing the coefficients on the MT^{th} record.

MT Number of set of coefficients being put onto disc by subroutine REDUCE.

MY Number of record of real data being put to disc at the end of FILE.

MZ As MY but for integer data.

NCORN Number of corners in the whole structure.

NDF The global reference number given to a degree of freedom which is staying active after the end of REDUCE.

NEX The number of external degrees of freedom and at times includes the other freedoms which are on a panel.

NEXC Is a count of those extra degrees of freedom being counted by NEXE but which have already included in the total by the effect of copying data assigned by a previous panel.

NEXE Is a count of the extra degrees of freedom required to take account of the relative slope of adjacent panels.

NG Number of springs in the whole structure.

NILOAD Number of loads being applied to the whole structure.

NPANEL Number of individual panels forming the whole structure.

NSH Equal to the number of sides of the plate element.

NSP Number of springs attached to a particular panel.

NSTX Number of stiffeners on a panel running parallel to the panel's local x axis.

NSTXLL Number of stiffeners on a previously processed panel which run parallel to that panel's local x axis.

NSTY As NSTX but parallel to local y axis.

NSTYLL As NSTXLL but parallel to local y axis.

NT Number of blocks of data used to define an element's thickness.

NTOT	The total number of degrees of freedom remaining active in the representation of the structure at that stage of assembly.
NTRIAS	Indicates the type of element assembly to be used at various stages during the formation of the stiffness matrix of a triangular panel.
NXBAYS	The number of bays formed by pairs of adjacent stiffeners running parallel to the local y axis and includes the bay formed between the local y axis and the first parallel stiffener and similarly between the last stiffener and adjacent edge.
NYBAYS	As NXBAYS but for stiffeners running parallel to the local x axis.

1. Introduction.

1.1 General outline.

1.1.1 This work combines both a general approach to finite element analysis of very large plated structures and a detailed study of the examination for structural strength of the design of the cargo tanks of ships which carry cargoes of liquefied petroleum gas in bulk. To a large degree this work is more associated with computer program coding than with finite element theory.

1.1.2 Though classification societies have in recent years developed their own approaches to such examinations, there still remains work to be done in this area. In a foreword to the study on gas tank loading criteria by Bass et al (1), Rear Admiral Benkert W.M. of the U.S. Coast Guard wrote "These tanks pose design and fabrication problems that were not previously encountered". It is interesting, that at about the time that foreword was being written in America, in this country the need for an independent method for examining the structural designs of gas tanks and their supports was identified by the Surveyor General of the Marine Division of the Department of Trade (2).

1.1.3 This theme of concern about the structures of gas carriers also appears in the reports of the International Ship Structures Congresses. In the opening remarks of the report of the 1976 I.S.S.C. (3), Merega commented on their previous meeting in Hamburg (4) and noted that, then they had said container ships and large tankers presented the greatest problems relating to stress distribution in hull structures and that for future researches, particular care needed to be devoted to the structural problems of special type ships, more and more common in practice, such as liquefied gas carriers, for which only now was there an attempt to set up calculation methodologies.

1.2 Pertinent shipbuilding rules.

1.2.1 Ships have been built to comply with the various sets of rules for generations. When Archimedes worked on the solutions to problems relating to Greek galleys, he was updating what were in effect the then current construction rules. Construction rules became more formal, when in 1834 Lloyd's Register of Shipping published a set of rules for the construction of wooden sailing ships, though those rules did not have statutory authority. Rules were and still are written and modified in the light of experience. So that over the years a fairly foolproof set of rules can evolve. This was a satisfactory system when ships were by present standards quite small and no great penalty to speed or cost was incurred by making plating a little thicker than was necessary to allow for the occurance of any higher than anticipated stresses. Special efforts are made nowadays to get hull plating as thin as possible. In a large tanker a reduction of 10% in plate thickness may save up to 3,000 tonnes. New designs and concepts are appearing with which the old proven rules give thicknesses too great for good economics or the dimensions derived for some panels may be too large or the rules are just unable to cope. Such a situation existed with ships which carry liquefied petroleum gas in bulk. Liquefied petroleum gas is potentially a very dangerous chemical, some of its hazards are given in an article in the New Scientist (5). Every effort must be made by the design team to see that as far as possible their design will give a vessel which can carry this cargo in safety.

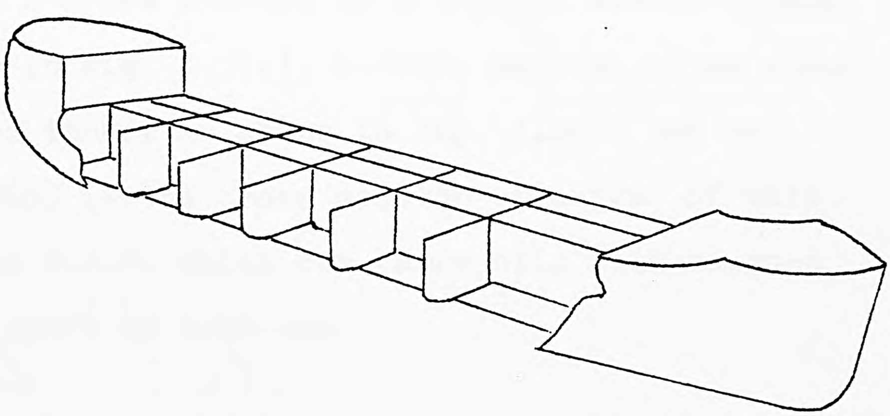
1.2.2 A special set of rules prepared by the International Maritime Organization apply to the construction of new tankers carrying liquefied gases in bulk (6). A second set of rules (7) by the same body apply to those tankers which are not new. Ships which are contracted for after 31 October 1976 or are delivered after 30 June 1978 are considered to be existing and are therefore not new. These are not highly detailed rules of construction but are principles to follow and standards to achieve. So that there is scope left for a designer, if he keeps within defined guidelines.

1.2.3 A list of chemicals which are considered to be liquefied gases is published by the Department of Trade (8).

1.3 Ship structures.

1.3.1 It is necessary to explain a little of the construction of some types ships.

1.3.2 A tanker is a vessel designed to carry liquid in bulk. A cutaway view of a tanker is given in Fig. 1.1.



Typical tanker structure.

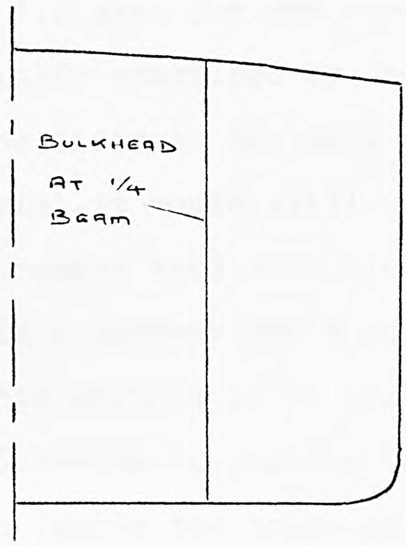
Fig. 1.1

Such a ship at the large end of the scale is a quarter of a mile long and can carry half a million tonnes of crude oil. Large crude oil tankers are called Very Large Crude Carriers (V.L.C.C.), such ships are the largest moving objects ever made by man.

1.3.3 Longitudinally, the shell of the hull is stiffened by longitudinals spaced about one metre apart and longitudinal bulkheads located at about a quarter of the full breadth from the sides. In addition to providing longitudinal strength these bulkheads reduce the transverse shift of the cargo. Transversely, the hull is stiffened by web frames four to five metres apart and subdivided by oil tight bulkheads fitted at intervals of about 20% of the ship's length. The tanks are fitted with at least one transverse perforated wash bulkhead per tank. The purpose of the wash bulkheads is to provide support for the bottom structure and to reduce the sloshing loads.

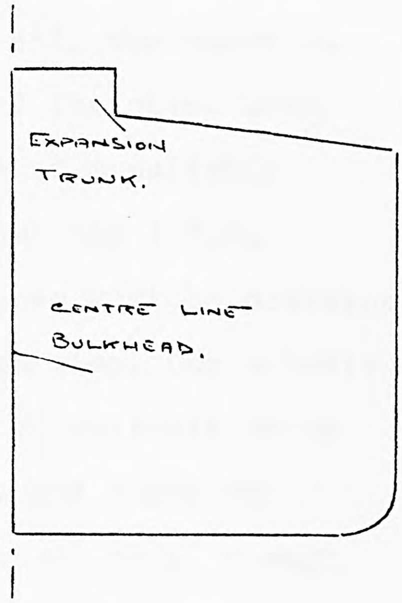
1.3.4 A cross section of a typical modern tanker is shown in Fig. 1.2(a). A cross section of an older design of tanker is shown in Fig. 1.2(b) and in Fig. 1.2(c) is the cross section of a type of ship called an O.B.O. which can carry oil, bulk cargoes such as grain or iron ore.

1.3.5 The type of ship which is of interest in this work is that group of tankers which carry liquefied petroleum gas in bulk. Ships are in general built of mild steel. It is cheap and easily worked. L.P.G. is liquid when at or below $-47^{o}C$ for atmospheric pressure. At such a temperature mild steel will be subject to brittle fracture in ordinary use. Brittle fracture



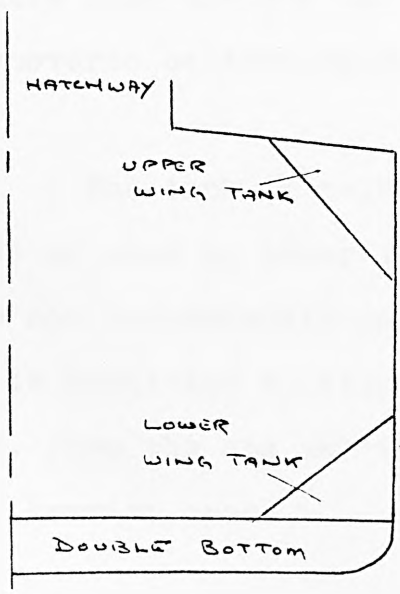
Modern tanker section

Fig. 1.2(a)



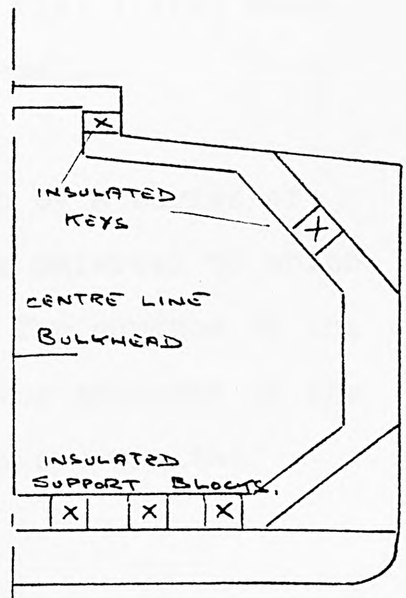
Older tanker section

Fig. 1.2(b)



O.B.O. section

Fig. 1.2(c)



L.P.G. carrier section

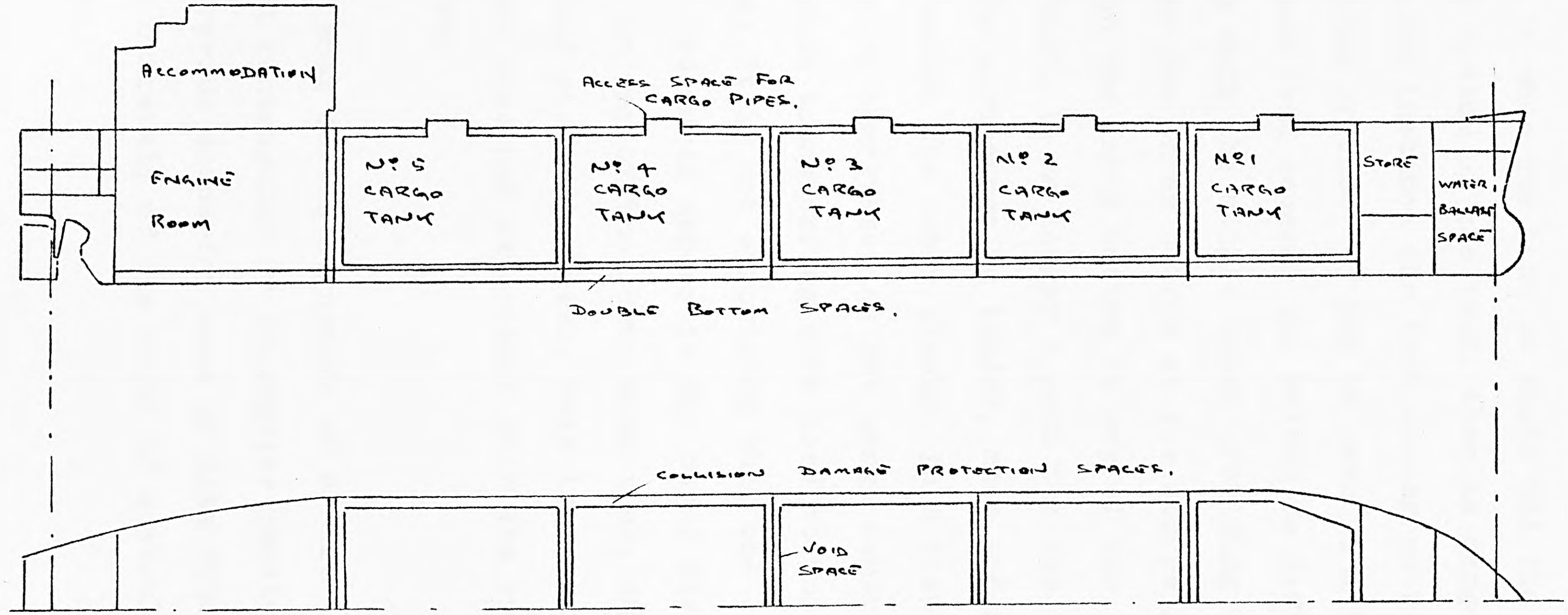
Fig. 1.2(d)

in ship's structures, which made a dramatic appearance during World War Two is now a well documented subject and needs no comment here (9). It can be seen from Fig. 1.2 that for the carriage of oil, the cargo is partially contained by the sides of the ship. Even if the sides of the ship were made of a suitable material it would still not overcome the I.M.O. requirement that such liquefied gases must be contained within a primary and a secondary barrier. One solution to this problem is to build tanks of suitable steel or aluminium to contain the L.P.G. and carry the tanks inside the holds of a mild steel ship, though not allowing the hull and the tanks to be in thermal contact.

1.3.6 The design which has developed for this type combines the cross section of an O.B.O. with a centre line divided oil tanker. Fig. 1.2(d) shows a transverse section of such a vessel.

The tank is held in place by a series of blocks of wood or other insulating material to which it is not mechanically connected. The surface of the tank is insulated to stop heat being absorbed by the L.P.G. from the sea and the atmosphere via the inter barrier space.

1.3.7 The structural design of the ship is such that the structure of the ship under a tank is more flexible than the tank. This gives rise to movement



Length overall 210. m Breadth moulded 31.4m Loaded draught 10.8 m Cargo 39500 tonnes

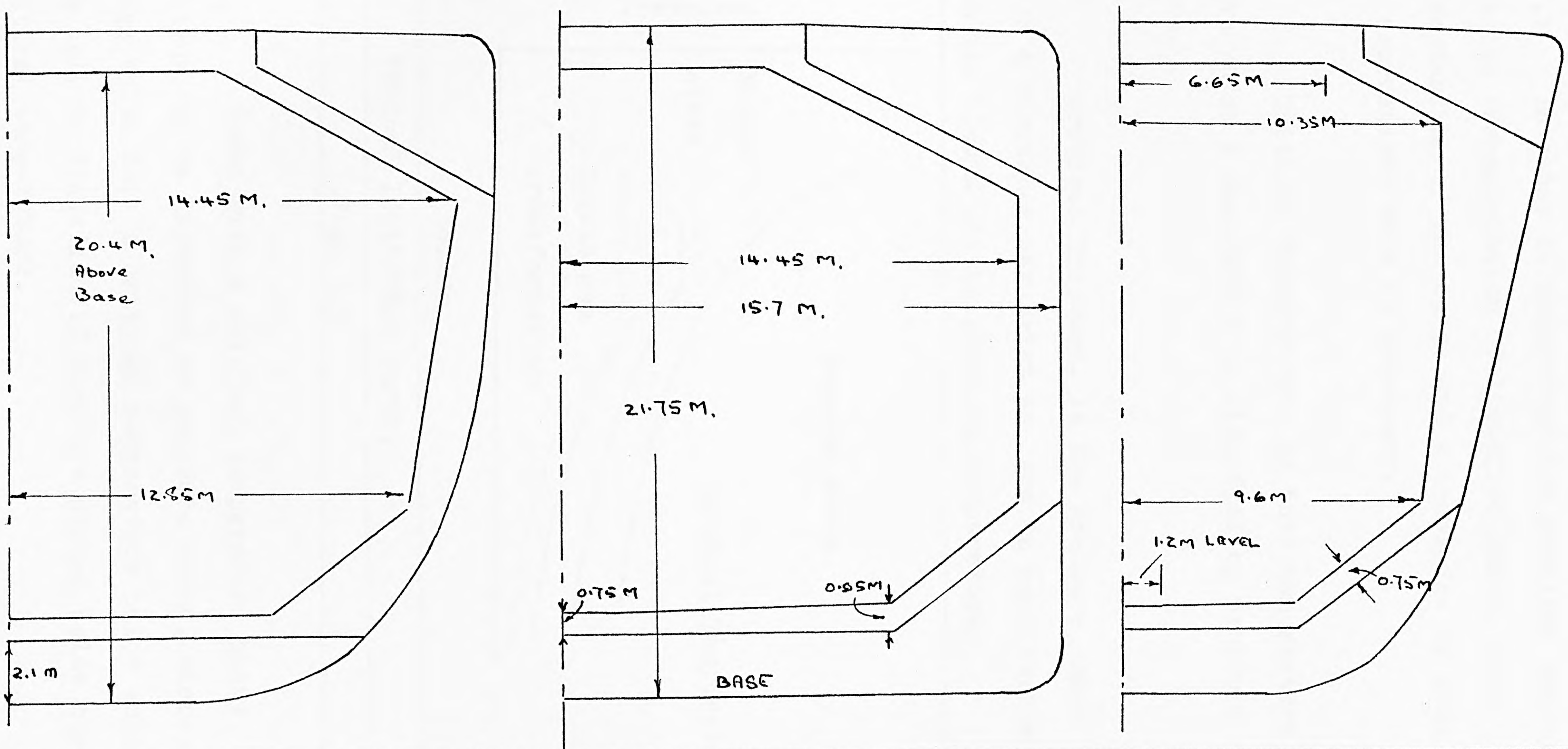
Longitudinal section through the hull and cargo tanks of a typical L.P.G. carrier.

Fig 1.3

of the tank bottom relative to the top of the insulating blocks. The cause of this can be seen as follows. If when the tank is empty all the blocks touch the bottom of the tank, then as cargo is loaded in the first instance the load coming on to the ship's inner bottom or tank top may be seen as a uniformally distributed load causing the bottom to deflect downwards with the ship's sides providing the reaction. But now as the inner bottom at the centre line is lower than the inner bottom in way of the outer support blocks, there is no longer a path for the load at the centre. As more cargo is loaded, more and more load is passed through the outer blocks. In a seaway this situation is aggrevated as the cargo centre of gravity moves across transferring more load to one side than the other. It is not surprising that one of the regions in which fractures appear is the outer edge of the ship's tank top in way of the lower wing tank, an obvious point of high and changing stress. This is an example of an item where time and experience show the rules needed to be changed.

1.3.8 Most of the structure of a ship is orthoganally stiffened rectangular or triangular panels. This should provide scope for ease of data generating, which is currently of the order of several man-months per tank.

1.4 Definitions and notes relating to liquefied gases.



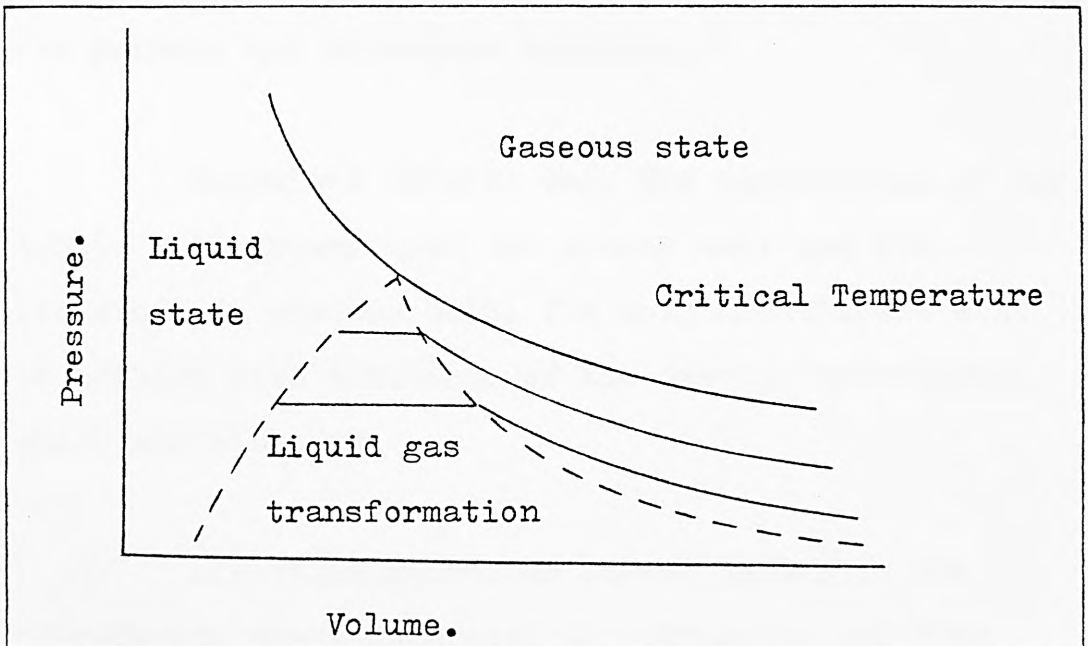
Tank and hull sections of the L.P.G. carrier shown in Fig. 1.3

Fig. 1.4

1.4.1 In order to understand the problems associated with the transportation of liquefied gases, some understanding of the cargo and a knowledge of some of the expressions used is necessary.

Critical Temperature. Is that temperature above which a gas cannot be liquefied by pressure alone.

Critical Pressure. Is the pressure under which a substance may exist as gas in equilibrium with the liquid at the critical temperature.



Typical Isothermic Curve.

Fig. 1.5

Gases with a critical temperature above ambient can be liquefied by pressure alone, whereas gases which have a critical temperature below ambient can only be liquefied if they are cooled below the critical temperature.

Primary Barrier. This is the inner element designed to contain the cargo when the cargo containment system includes two boundaries.

Secondary Barrier. This is the liquid resisting outer element of a cargo containment system designed to afford temporary containment of any envisaged leakage of liquid cargo through the primary barrier and to prevent the lowering of the temperature of the ship's structure to an unsafe level.

Inter Barrier Space. This is the space between the primary and secondary barriers.

Liquefied Natural Gas. The composition of any L.N.G. will depend upon the source well and the liquefaction process used. Its main constituent will be methane with additions of the heavier hydrocarbon gases and nitrogen.

Liquefied Petroleum Gases. Refers to the hydrocarbon gases separated at refineries and from the oil and gas wells which are gases at normal atmospheric temperatures and pressures, but which can be readily liquefied by pressure or refrigeration; propane and butane are saturated gases, propylene, butylene and butadiene are unsaturated gases. The saturated gases are mainly used for heating, motor fuel and aerosol propellant. The unsaturated gases are mainly used for chemical feedstock in the production of plastics.

Ethylene. Ethylene is used mainly in the production of polyethylene plastic, of ethylene oxide as a chemical intermediate and for the manufacture of ethylene glycol for anti-freeze.

Anhydrous Ammonia. Large quantities are used annually in the production of fertilizers. It is also used in the production of nickel and cobal, explosives and a chemical feedstock. Though not a petroleum gas it is suitable for transport by L.P.G. carriers.

1.5 Liquefied gas ship types.

1.5.1 From a consideration of the characteristics of the gases mentioned it can be seen that liquefied gases may be carried in various states from fully pressurized at ambient temperature to fully refrigerated at atmospheric pressure. These states may be classified as:

1. Under pressure at ambient temperature.
2. Under partial pressure at ambient temperature.
3. Under partial pressurization and full refrigeration.
4. At atmospheric pressure and fully refrigerated.

In case 4 there is a slight pressure maintained in excess of atmospheric.

Properties	Natural Gas	Ethylene	Propane	Ammonia	Ethylene Oxide	Butadiene 1,3	Butane
Carriage Temperature ^{o}C	−165	−104	−43.	−34.	−11.	−5.0	−1.0
Carriage Pressure Kgf/cm^2 Abs	1.04	1.04	1.04	1.04	0.45	1.04	1.04
Specific Gravity of Liquid	.474	.570	.583	.683	.913	.647	.602
Viscosity of Liquid Ns/m^2	.142	.125	.216	.174	.320	.166	.241
Specific Heat of Liquid Kcal/Kg/^{o}C	.465	.674	.500	1.070	.268	.518	.467
Latent Heat of Boiling Kcal/Kg	124.	115.	101.	326.	135.	99.0	91.0
Vapour Density relative to air	.554	.975	1.55	.597	1.52	1.88	2.90
L.E.L. to U.E.L. % by vol	5.3–14	2.7–28	24–9.5	14–28	3–100	2–11.5	1.8–8.5
Flash Point ^{o}C	very	low	−105	vague	−57.	−60.	−60.
Ignition Temperature ^{o}C	595.	450.	470.	652.	429.	450.	406.
Critical Temperature ^{o}C	−82.1	9.90	96.8	132.	196.	152.	152.
Critical Pressure Kgf/cm^2 Abs	45.8	50.5	42.0	112.	71.0	42.7	37.5

Characteristics of typical products carried by liquefied gas tankers.

Fig. 1.6

1.6 Types of gas tanks.

1.6.1 Tanks for the carriage of liquefied gases are designated by I.M.O. as Integral, Membrane and Independent.

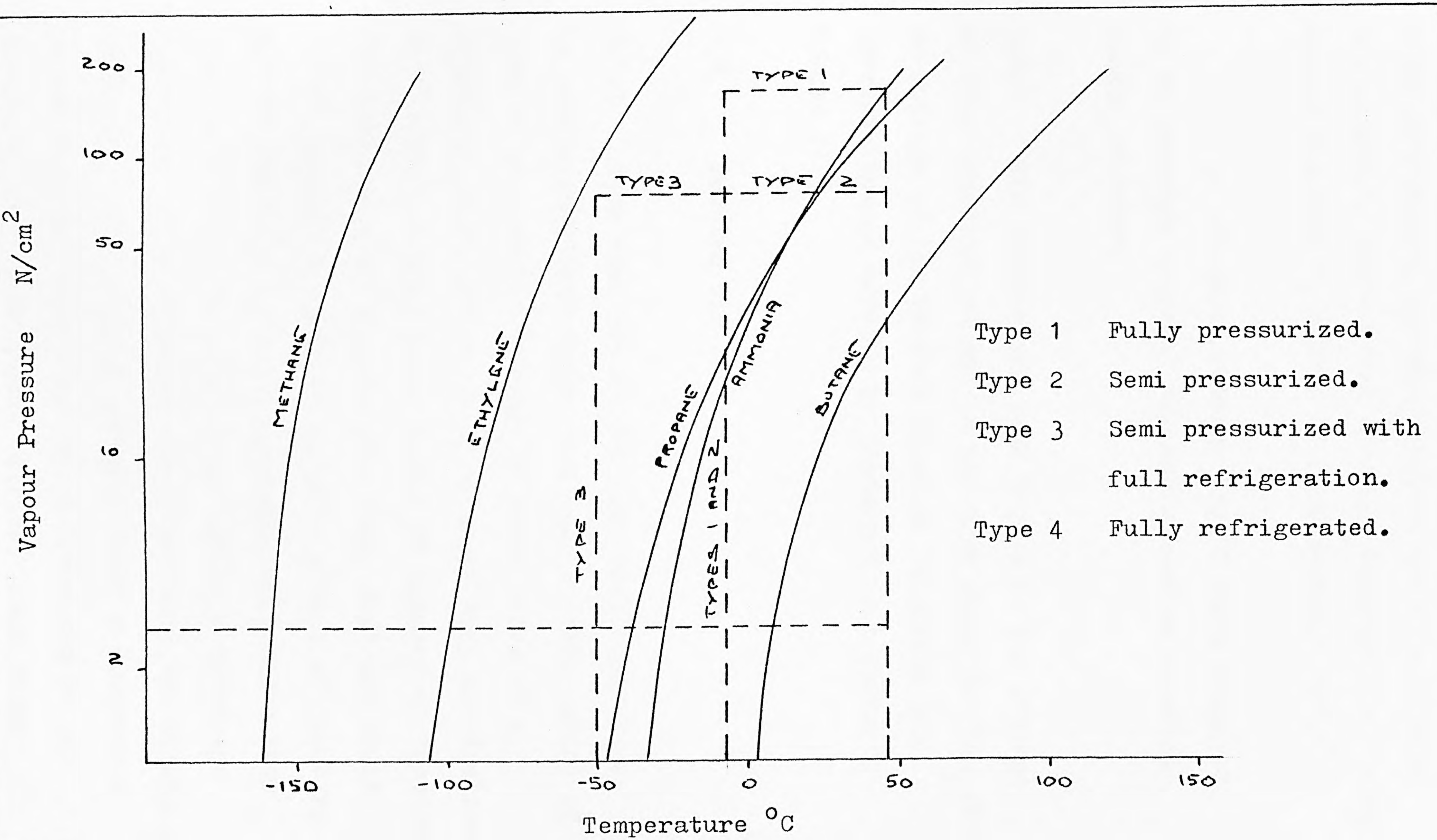
An integral tank is one which forms part of the structure of the ship and is influenced in the same manner and by the same loads which stress the adjacent hull structure.

Membrane tanks are non self-supporting tanks which consist of a thin layer supported through insulation by the adjacent hull structure.

Independent tanks are self-supporting, they do not form part of the ship's hull and are not essential to the hull strength. The three categories of independent tanks are:

1. Independent tanks type A which are designed primarily using recognised standards of classical ship structural analysis procedures. Where such tanks are primarily constructed of plane surfaces (gravity tanks), the design vapour pressure P_o should be less than 0.7 kg/cm^2.

2. Independent tanks type B which are designed using model tests, refined analytical tools and analysis



Liquefied Gas Ship Types. Fig. 1.7

methods to determine stress levels, fatigue life and crack propagation characteristics. Where such tanks are primarily constructed of plane surfaces the design vapour pressure P_o should be less than 0.7 kg/cm^2.

3. Independent tanks type C (also referred to as pressure vessels) are tanks meeting pressure vessel criteria.

1.6.2 The independent tank type B is the subject of this dissertation and is the type shown in Fig.1.2(d). An outline of the various types of liquefied gas carriers now in service is contained in a paper by Ffooks (11).

1.7 Problem size.

1.7.1 From what has so far been described it can be appreciated that there is a considerable amount of data to be produced for any of these types if a realistic idealization is to be used. Such an idealization will contain a very large number of degrees of freedom. The data required is increased above what one would expect because in general there is a lack of symmetry. The thickness of material increases with depth from the top of the tank, destroying symmetry about any horizontal plane. Forward and after tanks are shaped so that there is a lack of symmetry about a transverse plane. The only axis about which there can be any symmetry, the longitudinal vertical plane lacks structural symmetry when transverse loads, resulting from rolling destroy reaction symmetry. Strictly speaking even this last plane is not a plane of symmetry since stiffeners on the longitudinal centre line bulkhead are all on one side of the bulkhead.

1.8 Certification.

1.8.1 The Department of Trade, the authority who in the United Kingdom is responsible for issuing the Certificate of Fitness required by a gas ship, derive their authority from various Merchant Shipping Acts of Parliament (12). The principal, but not the first being dated 1894. Section 438 of that act refers to putting a mark on the side of the ship indicating the maximum draught to which it may be loaded. This mark is the Load Line mark incorrectly called by some the Plimsoll mark. Load Line Rules contain the statutory details of strength requirements for merchant ships. The Board of Trade, then the Government department responsible for civil shipping matters first issued Load Line Rules in January 1886, they were the same, with only slight amendments, as those previously prepared by Mr. Benjamin Martell and adopted by Lloyd's Register of Shipping in July 1882. They were in operation for 46 years and formed the basis of regulations adopted by other maritime nations. In those rules of a century ago little more than a check on the I/y value at midlength was all that was required as a strength check. Over the years the amount of examination has increased so that now the Department has to be satisfied inter alia as to the details of the structural design of the tanks and their supports in liquefied petroleum gas tankers.

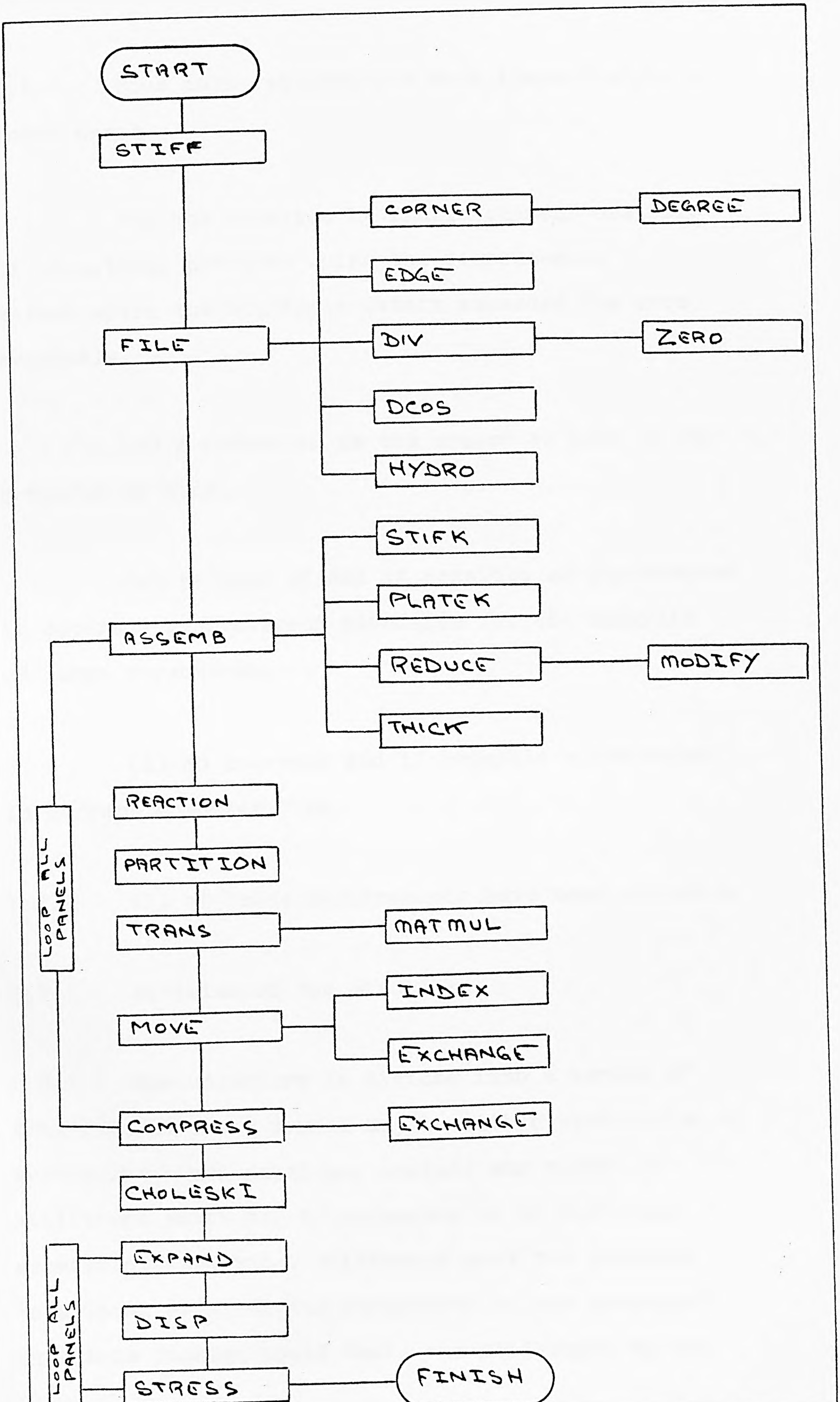
2. The problem and a suggestion for its solution.

2.1 Small facilities often available.

2.1.1 By the standards of past ship design, the techniques of today are very sophisticated using large main frame computers and teams of specialists. At the begining of the introduction it was written "This work relates to the examination of...", the words "This work relates to the design of..." was not written. The role of the examiner or checker of a finished design is in a rather poor relation position. Where for the design there may be one or more teams involved, for the checking there may be only one or two individuals with a limited amount of time and a small core computer or just part of a main frame. In the discussions after the reading of the paper on H.M.S. Invincible by Honner and Andrews (13), it was said that a large part of the professional staff involved in the design took about 18 months to prepare the finite element analysis data.

2.1.2 The intended objective of this work has been to improve the lot of the few who examine a completed design. The solution proposed here is applicable to most structures whether or not they are floating. Fig. 2.1 shows a block schematic of the package.

2.2 Four main requirements.



Calling order of subroutines

Fig. 2.1

2.2.1 Four main requirements were identified, these were:

(a) The solution by finite element analysis of structural problems using the displacement method where the stiffness matrix exceeded the core available.

(b) A reduction in the amount of data to be prepared by hand.

(c) No loss of and if possible an improvement in accuracy over current standards for the analysis of large structures.

(d) No increase and if possible a reduction in current computer time.

2.2.2 All of these requirements have been achieved.

2.3 Division of the structure.

2.3.1 The structure is divided into a series of flat panels. These panels may be either rectangular or triangular. Each panel may include any number of stiffeners which may if necessary be of differing geometric properties. Stiffeners must run parallel to a local axis. In the structures it was envisaged that this package would deal, the stiffeners do run parallel to a side. Appendix A lists the information required as input data for the program to run.

2.4 Forming the stiffness matrix.

2.4.1 The stiffness matrices of panels are prepared in the order of data submission. A process has been developed by which selected stiffness coefficients are removed so that the final stiffness matrix is only expressed in terms of those degrees of freedom on its perimeter plus any internal restraints and spring endings. This technique ensures that only the finally occupied space is almost all the space used. This is unlike the usual methods of substructuring (14) that require the core space for the full substructure before partitioning to express the matrix in terms of a lesser number of coefficients. This represents a large saving in the core space needed. Before adding the current panel's reduced stiffness matrix to the remainder of the matrix of the structure so far seen, input data relating to restraints, spring ends and the orientation of adjacent panels is scanned and the current panel's stiffness matrix modified as necessary to make it complete in itself, whilst being compatible with adjacent panels.

2.4.2 After adding the stiffness matrix of the current panel (small k) to the stiffness matrix of the whole structure so far processed before the current panel's appearance (large K) to form a new large K, more space is made available in the core for the next panel's stiffness matrix. This space is arranged irrespective of whether or not there is sufficient available, by compressing the existing large K stiffness matrix. This compressing is achieved by removing those degrees of freedom which are no longer on the wavefront. There is a saving in core space required over the well known wavefront method, this is because the sparcity usual with large bandwidths does not exist. The wavefront method is amongst the procedures contained in Meyer's paper (15). This lack of sparcity is a natural feature arising from the previous reduction routine and not an engineered arrangement such as a skyline technique (16). Lack of sparcity means that programs such as that by Cheung and Khatua (17) which is based on not storing zero submatrices, will not run as designed and therefore will not be at optimum efficiency.

2.4.3 This stage of making the stiffness matrix smaller can be superficially likened to the first reduction stage where its plate and bar elements were replaced by super-elements formed of the reduced panels. This analogy will not stand up to indepth comparison as the programing is completely different and simpler in the second case, as the whole stiffness matrix exists when the compressing routine starts. This reducing and compressing the wavefront continues right through the structure until all the panels have been processed. Before inversion of the very small remaining stiffness matrix takes place the coefficients due to the insulating support blocks are added.

2.5 Loads.

2.5.1 All reasonable loading conditions are allowed. The principal loadings coming on the structure are hydrostatic, thermal and acceleration including gravity. In these cases it is sufficient to put the key parameters in the data and the load vector will be formed automatically. Individual loads may be applied as required and there is no limit on their number.

2.6 Springs

2.6.1 This package has available three types of spring, these are described below.

2.6.2 The first type of spring is a representation of the thermally insulating block which is incapable of transmitting a tension load.

2.6.3 In the consideration of a freely floating body there are no restraints on any degrees of freedom. Each degree of freedom has a potential for displacing due to elastic deformation and to rigid body displacement. For a set of constant loads corresponding to static equilibrium, the rigid body displacement will be indeterminate and the stiffness matrix will be singular. This singularity situation can be removed by using the self defining springs, which are the second type available in the package. These springs are attached at one end to suitable freedoms delineated by the analyst and at the other end to undefined hypothetical foundations.

2.6.4 It may be that the whole or symmetrical part of the structure is not to be represented in the analysis. In this case the influence of adjacent structures on that part to be idealised must be allowed for. This can be done by estimating the stiffness of adjacent structures, they can then be represented by a series of springs attached to hypothetical foundations as those restraining free body motion, but in this case these third type springs are not self defining but use the value estimated by the analyst.

2.7 Post inversion outline.

2.7.1 Immediately after inversion the resulting displacements are scanned and the final stiffness matrix modified to remove the coefficients of any spring representing an insulating block which has gone into tension. This does not apply to those springs attached to hypothetical foundations which can handle tension and compression. This process is repeated until all those springs not in compression have been removed from the stiffness matrix.

2.7.2 The remaining part of the process now deals with the panels in the reverse order to that which applied during the assembly routine. The first step is to form all the displacements of degrees of freedom on the current plate's perimeters together with internal restraints of those freedoms which were treated as external freedoms. This is started by the recovery from disc of the inverse of the stiffness matrix of all such degrees of freedom, the stiffness matrix of the cross terms between such freedoms and those freedoms which were removed. The displacements are then derived by operating on this recovered information and the known external displacement values.

2.7.3 Subsequent to the formation of all the displacements of a panel's external or psuedo external degrees of freedom, the process of generating the values of the internal freedoms' displacements is begun. This is achieved in subroutine DISP by substituting known displacement values into the equations, the coefficients of which were stored onto disc during the panel's stiffness assembly and reduction stages. However these stages were run entirely with the panel related to its own local axes, wheras some of the displacement values available to seed this reconstruction are related to global axes. Therefore a selective transformation operation must be conducted to change the orientation of those displacements which are not suitably aligned.

2.7.4 The final stage of generating the stresses of the plates and beams is handled by subroutine STRESS. Plate stresses are printed out for the plate centre after meaning the stresses calculated for each corner of the rectangular plate elements. In the case of triangular elements because they are constant strain elements, meaning the corner values is superfluous. For beams, the stresses are given for the outer fibre of the beam edge having the greatest absolute value of stress.

2.8 Savings.

2.8.1 From what has already been written it can be appreciated how large core operations can be done in a very small space, Much time is saved because the final matrix which may need to be inverted several times to find the final supporting block arrangement is so small. The converse does not however apply, in that if there are no springs the time does not go up automatically when compared with the processing of the entire structure within one large core. This is due to all the prior manipulations which form a continuous partial inversion of the overall matrix.

2.8.2 Accuracy around stiffeners is improved because more core space is now available and the practice of lumping several stiffeners into one where they occur in very large structures may be dispensed with.

2.8.3 The possible though not necessary loss of accuracy due to using only simple plate elements in this development of the work can be balanced against the ability to use more elements. Though it is shown in chapter 11 that this technique may be used with elements having more degrees of freedom than just those of the four node rectangular and three node triangular elements used here.

2.8.4 Much work has been put into data generating from limited hand prepared information and use has been made of the repetative nature of many large structures. However structures having a completely random formation can be dealt with. All element dimensioning, node co-ordinate values, node numbering and manipulation and non individual loadings have been made fully automatic. This has led to a considerable saving in hand prepared data over that required for the analysis packages available at The City University. These packages are Lusas (18) and Sap IV (19).

3. Data generating and minimising input.

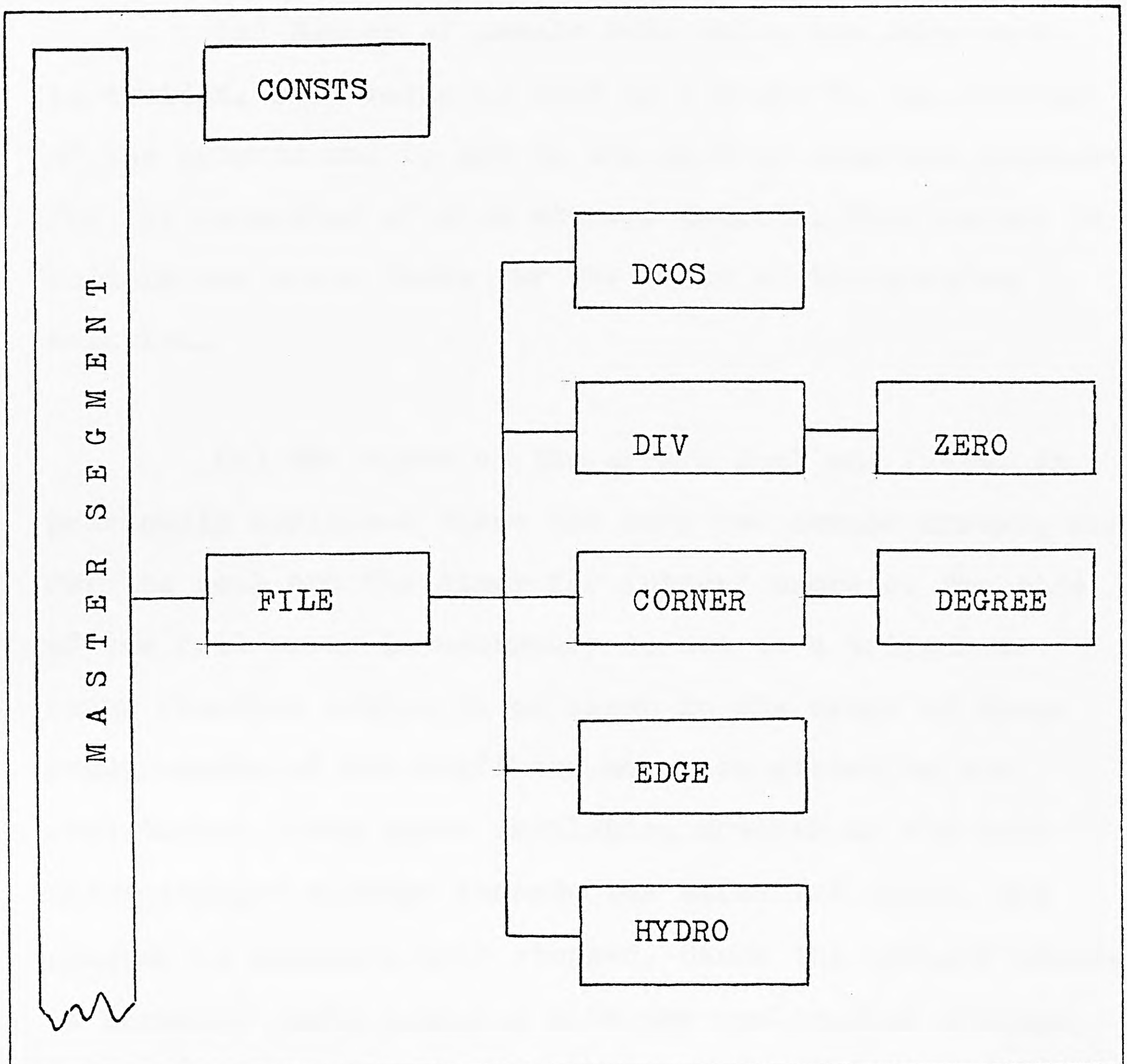
3.1 Generating outline

3.1.1 To give meaningful results to a program for the analysis of large structures, requires a considerable amount of data. One of the targets of this work was that the hand preparation of data should be kept to a minimum. This has caused a considerable part of the program being used for the generation of data from skeletal input.

Data for the solution of this problem is divided into two main groups. That which is common to the whole problem e.g. material and stiffener properties and that which is peculiar to a particular panel e.g. dimensions and the number of stiffeners attached. There is a small third group containing details of the physical characteristics of the linear springs but this information is not read in until the final matrix is ready for inversion and does not need any disc storage or data generating.

3.1.2 For the ease of program writing the required data is generated by a series of subroutines. One large routine would have been possible but rather cumbersome. Fig. 3.1 shows the calling order of those subroutines. Details of the form and order in which data must be given is listed in appendix A.

3.2 Description of each data generating subroutine.



Hierarchy of data generating subroutines.

Fig, 3.1

3.2.1 Subroutine CONSTS.

The first of the two main groups mentioned in 3.1.1 is dealt with by CONSTS. CONSTS has two main functions it reads and stores the information given below and then using that list generates and stores data required by other subroutines.

Below is a list of the constant value items read and stored by CONSTS:

(a) Number of panels into which the structure is divided. This value is used as a limit to the running of the program and to set up the size of memories required for the recording of disc storage details. This record is held in the array Idata for the whole of the problem solution.

(b) The sizes of the arrays Real and Idata. As previously explained there are only two common arrays, one for the real and the other for integer numbers. The size of the real array is necessary to act as a trigger to cause remedial action to be taken in the event of space requirements of the stiffness matrices exceeding the real number array space available. However in the case where integer storage exceeds the allocated space, the program is automatically stopped. Since the integer storage is normally small compared with the real number storage, a decision was made that it was not justified to include a subroutine for offloading to and recalling from disc, parts of the various indicies during the processing of individual panels. At the completion of each panel's stiffness matrix assembly all integers except the overall index are either abandoned or offloaded to disc, which should create sufficient integer space. If this does not then the integer array size may be enlarged at the expense of the real array and remedial action mentioned above is used to cope with any consequential lack of real number space.

It is not necessary to put the actual array sizes in the input data. Any number equal to or smaller than the actual array size will suffice. Not using the true values will cause earlier action to offload array storage. Whereas a number in excess of the true size may cause errors due to overwriting of data but not necessarily stop the program running.

(c) The number of corners in the whole structure. The technique used in this program for deciding if the degrees of freedom along any edge have become inactive, is to check the status of that edge's corners. A section of the integer array equal in length to the number of corners in the whole structure is set aside for this purpose. This action is described in the next section dealing with the subroutine FILE.

(d) Young's modulus and Poisson's ratio.

(e) The number of springs of all types.

(f) The number of different types of stiffeners to be used in the whole structure;

1. about which the geometric properties are known and which will be included in the input data and

2. about which the geometric properties are not known and will require to be generated from a list of dimensions which will be added later.

(g) The only nodes whose co-ordinates need to be specified during input are the panel's corner nodes.

The program contains a facility for the calculation of the co-ordinates of the intersections of a rectangular mesh. At this stage of the input a trigger value is supplied indicating whether the panel corner node co-ordinates will be read in individually or using the facility, generated as a mesh with the co-ordinates of any nodes which are required but do not fall on any of the intersections of the mesh being subsequently read in.

(h) The number of corner nodes which are to be read in individually.

(i) The dimensions necessary to calculate the cross sectional area, inertia and eccentricity of any stiffener whose geometric properties are not available for reading in.

(j) Information to define the mesh referred to in (g) above. This refers to each global axis in turn and contains the details of the number of groups of nodes at constant spacing along each side and the distance along the axis before the node spacing changes.

3.2.1.1 Generating geometric properties.

The cross sections of which, geometric properties are required are represented by a set of rectangles. The process of obtaining the properties is based upon the standard procedure of summing areas and moments. The flow chart at Fig. 3.2 gives the outline of this procedure. The

Flow chart for the generation of the geometric properties of stiffeners.

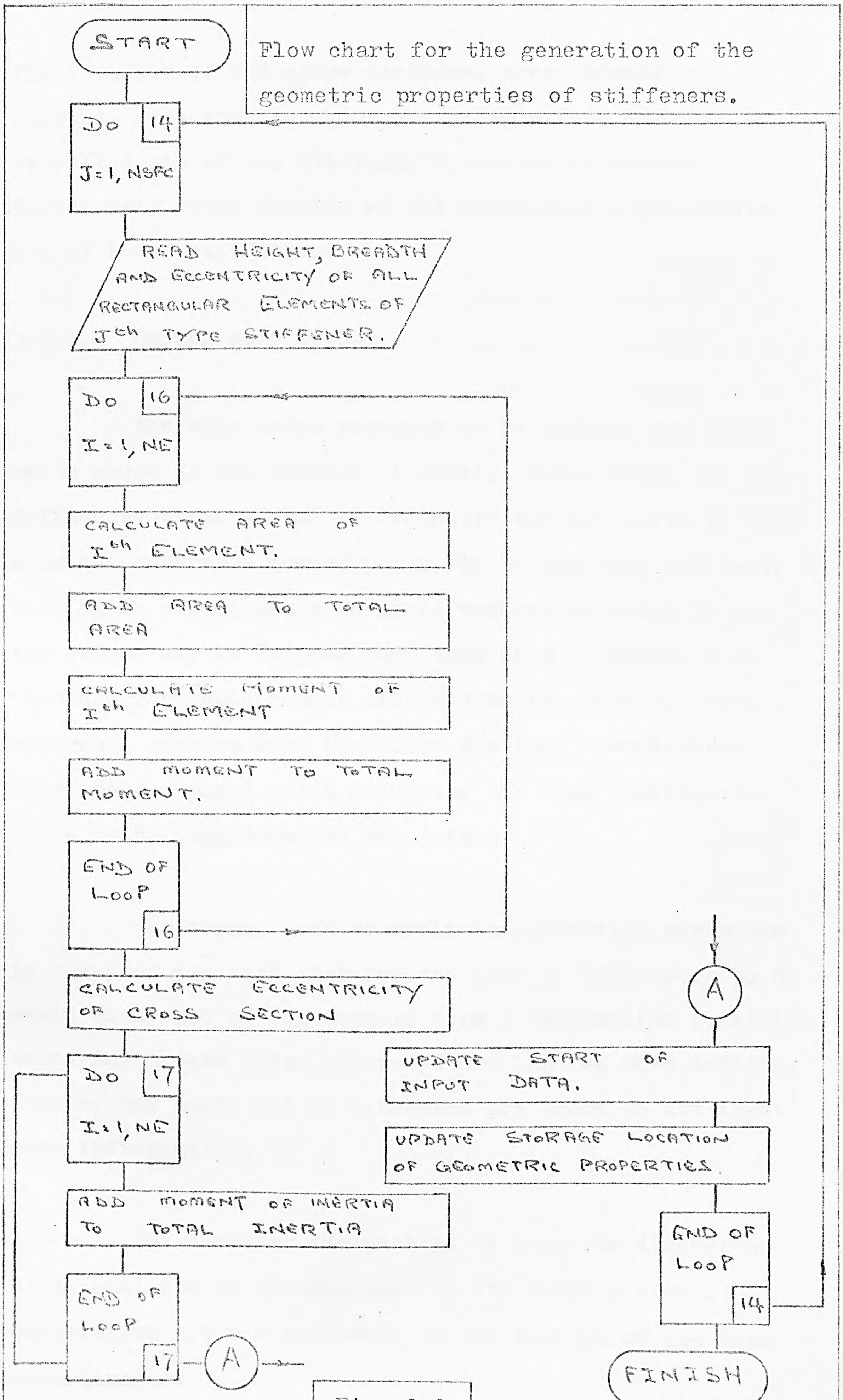


Fig. 3.2

final values of the cross sectional area, moment of inertia, eccentricity about the plate's node and the overall depth of the stiffener's section is stored immediately after details of the rectangles representing all of these stiffeners.

3.2.1.2 Corner node mesh.

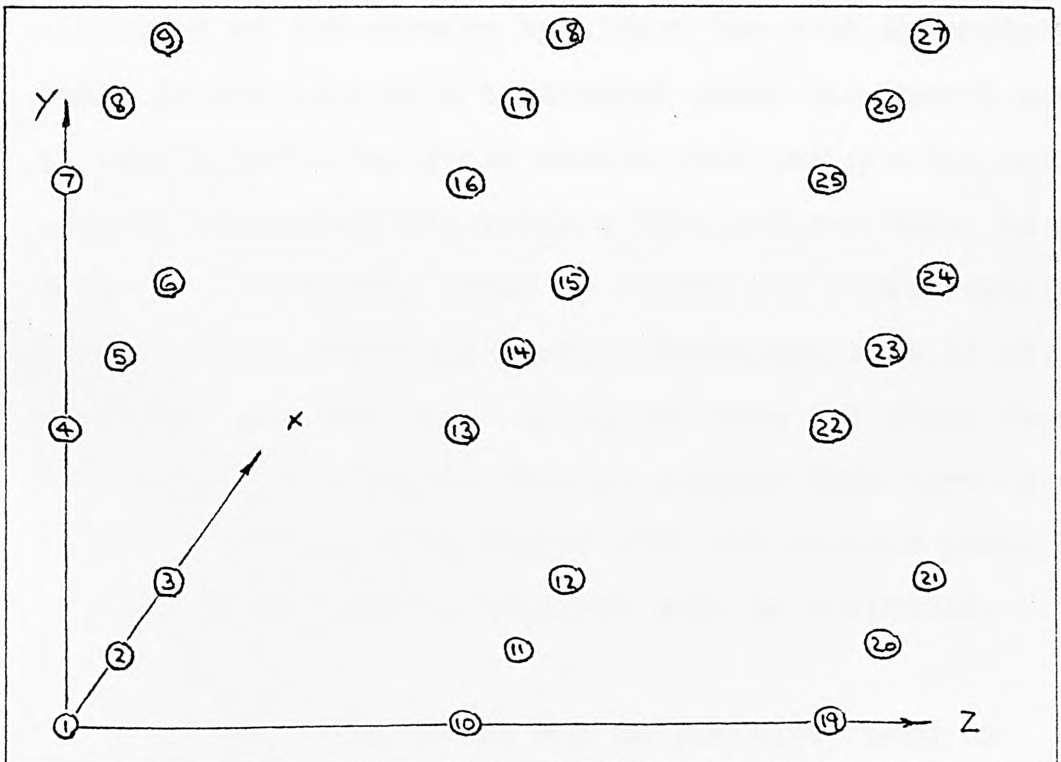
The only nodes required to be defined are those which occur at the corners of panels. Those nodes are defined by their global co-ordinates and are given by hand a unique reference number which may be entirely arbitary, bearing no relationship to those numbers adjacent to it. Any corner may be defined more than once by having more than one reference number assigned to it, each of these reference numbers must then have the same co-ordinates. This enables panel edges occupying the same position in space to be considered as not joined.

For corner node co-ordinate generating, advantage is taken of the fact that for the type of structures to be dealt with most of the corners form a rectangular prismatic mesh. Any excess nodes generated are ignored when defining panels. Any nodes not so generated are added to the input data individually.

Any need for the package to know the dimensions of a panel can be accommodated by its doing a simple calculation after a reference to the storage of the mesh co-ordinates.

The mesh is generated so that sequential calculating of co-ordinates of nodes proceeds along or parallel to the OX axis moving away from the ZOY plane. When node co-ordinates have been completed for the defined extent of the X axis the process is repeated starting from the next node on the ZOY plane to complete the next row of nodes parallel to the OX axis. Only when all the co-ordinates of a complete plane of nodes in or parallel to the XOY plane are calculated does the loop move to the plane parallel the XOY plane passing through the next Z co-ordinates.

The nodes are not directly numbered but since the nodes are dealt with in a definite pattern they are numbered by implication. The set of co-ordinates for each node is stored in the order of that pattern.



Node positions/axes orientation taken by co-ordinate calculating routine for a 3 * 3 * 3 mesh.

Fig. 3.3

3.3 Subroutine FILE.

3.3.1 The second of the two main groups mentioned in 3.1.1 is dealt with by this subroutine which is called once for every panel into which the whole structure is divided. As shown in Fig. 3.1 this subroutine controls directly the calling of the five subroutines which do the bulk of the data generating. Only when every panel has been processed by FILE does the process move on to start forming the first stiffness matrix.

3.3.2 Panel reference numbers.

Each panel is defined by five integers. In the case of a rectangular panel the first four being the numbers allocated to its corners by either the mesh generator or by hand. In the case of a triangular panel the fourth number is made a zero. The first number read defines the local origin, the second the local x axis and the third in the case of a triangular panel or fourth for a rectangular panel defines the local y axis. Numbering must be anti clockwise when the panel is viewed from the front face. ✗ The absence of a fourth integer greater than zero is used to inform various subroutines that the current panel is triangular and certain triggers must be activated.

The fifth number may be positive, zero or negative. A positive value will cause the input data for this panel to be stored. It may then be copied by any subsequent identical panel which uses a negative fifth

✗ Face is as in R.H. rule so that front face stiffener eccentricity is positive.

integer of the same absolute value as the original. A zero value allows data to be read, processed and then overwritten without storing.

Mesh generated and hand prepared node co-ordinates are stored in sequence, so that once the corner number is known the three real numbers defining it in space may be easily found and stored. At this stage the list referred to at 3.2.1(c) is prepared. This is done by increasing by one the contents of the location given by the node number, on every occasion a node number is read.

3.3.3 Disc storage of data.

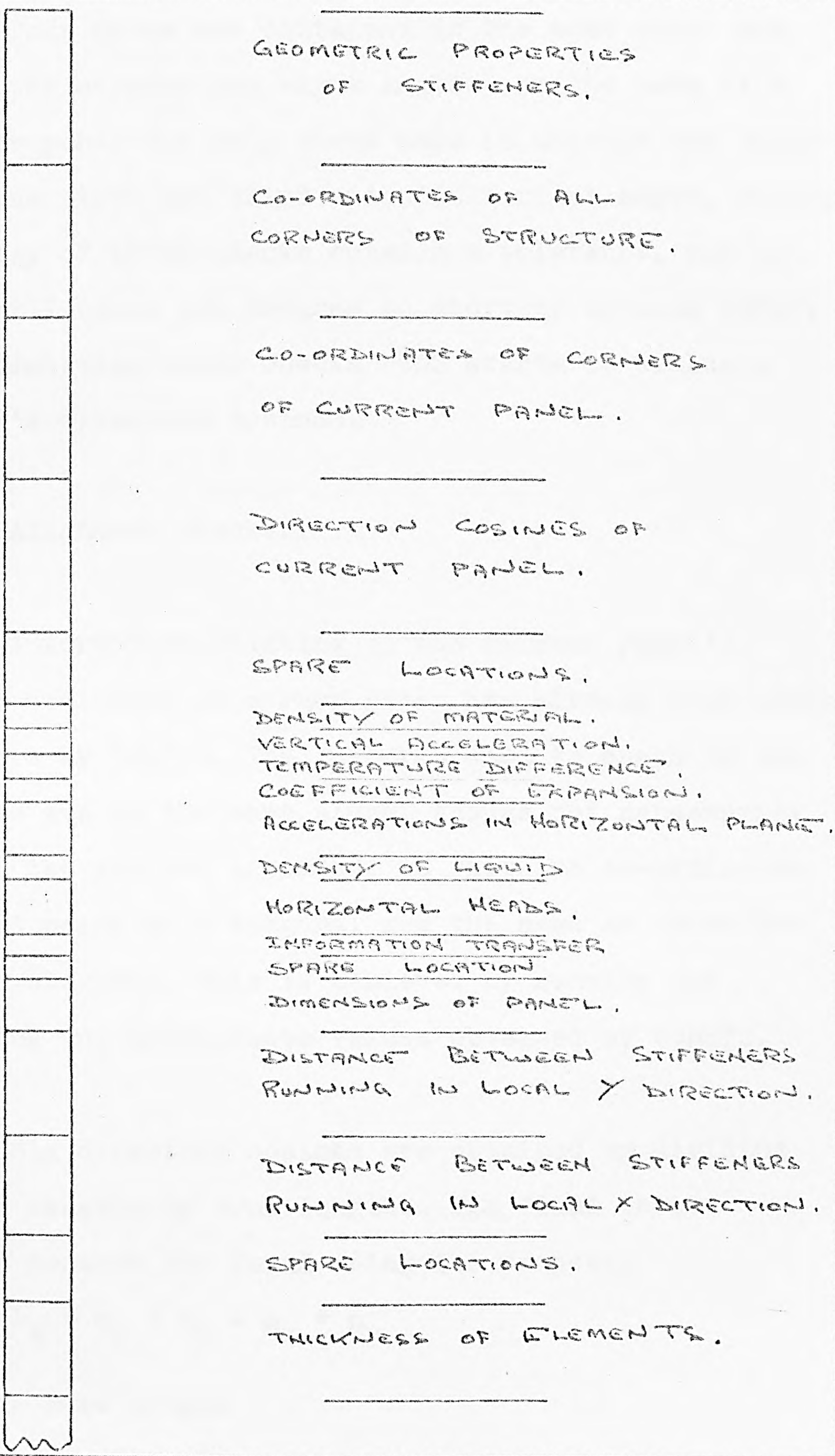
At the completion of each panel's processing by FILE the generated and some of the read in data is off-loaded to two disc storages, one for integer and one for real numbers. A list of this data and its relative storage locations is given in Fig. 3.4.

3.3.4 The other tasks undertaken by FILE.

The processing and generating details shown in Fig. 3.5 relating to material thicknesses, restraints and loadings (body, point, thermal, and hydrostatic) will be dealt with in later chapters.

3.4 Subroutine DCOS.

3.4.1 In the case of a rectangular panel, DCOS checks



Final locations of real number data in array Real at the end of subroutine FILE.

Fig. 3.4

that all four nodes are contained in the same plane and that all the corners are right angles. In the case of a triangular panel the only check made is whether the angle between the first and third sides is a right angle. Failure to meet any of these checks outside a tolerance, now set at 0.1% will cause the program to abort by calling FAULT. Before completing these checks DCOS starts to evaluate the panel's direction cosines.

3.4.2 Alignment checks.

Information relating to the current panel's global co-ordinates of corner nodes has already been stored in the core by CONSTS. The criteria used to check if the four nodes are in the same plane, though not necessarily the plane the analyst intended, is that the co-ordinates of the mid point of a diagonal are the same as those for the other diagonal. This is achieved by summing and subtracting the appropriate values obtained by CONSTS.

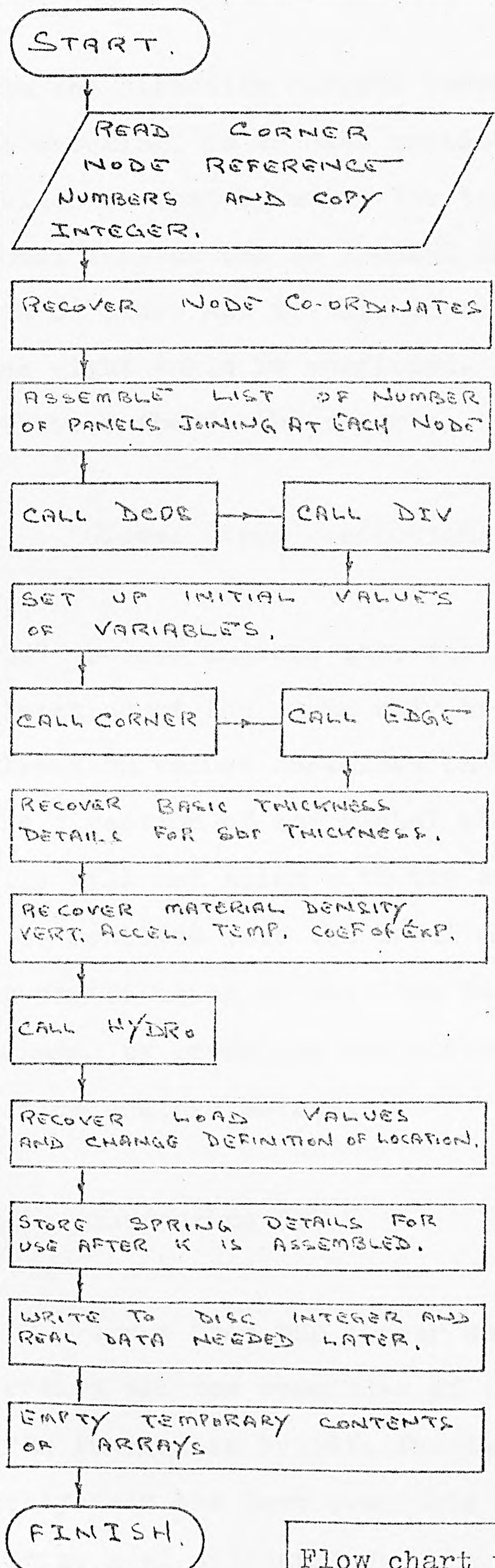
Six direction cosines are obtained by dividing projected lengths by true lengths. The final three direction cosines are found using the property

$$l_x = m_y * n_z - m_z * n_y$$

use of the more simple

$$l_x^2 + m_x^2 + n_x^2 = 1$$

fails because of the inability to put the right sign to the derived root value. The reason for starting to



Flow chart for subroutine FILE showing some stages not yet mentioned in the text.

Fig. 3.5

obtain the direction cosines before the finish of node value checking, is to have available the true length of the sides so that by using the theorem of Pythagoras the diagonal lengths can be checked in terms of the true length of sides and if this is true, the value of the corner right angle is confirmed. Failure to comply with the theorem aborts the program.

3.4.3 Global force resolution.

3.4.3.1 Forces induced into the structure due to acceleration of the whole body are evaluated using acceleration values contained in the input data related to the direction of the global axes. These forces which usually will not align with the appropriate local axis must be resolved into the local axes directions before their contribution to the load vector can be assessed. This is managed by inverting the already formed local axes direction cosines matrix.

3.5 Subroutine DIV.

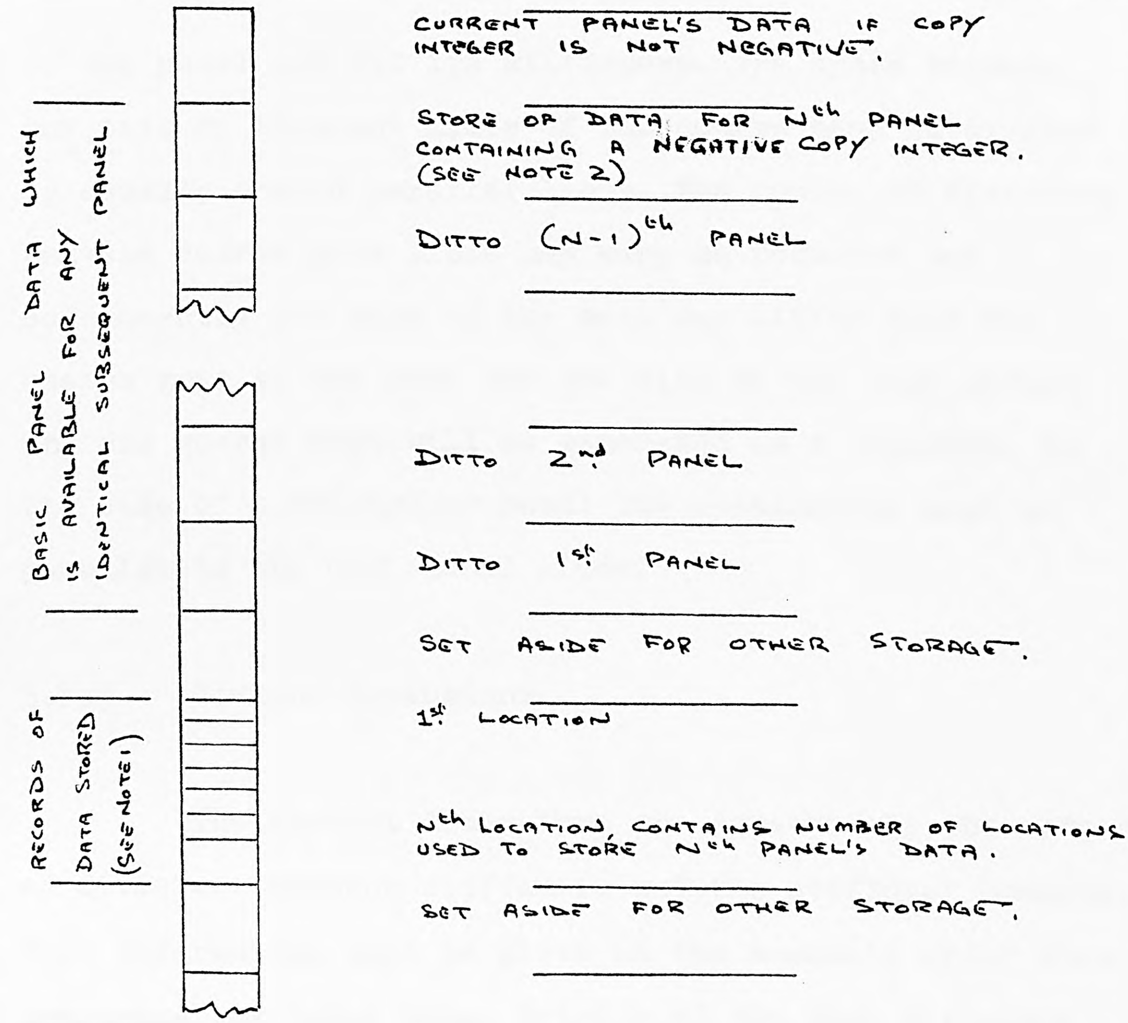
3.5.1 Apart from the linear spring stiffness properties all the remainder of the data required by the program is read in by DIV. The data is read in by DIV directly into the last available locations at the end of the array Real allocated for the storage of the stiffness matrix. Whether the data is put into this space or overwrites the last panel's set of data depends upon the value of the copy integer. The data for each panel follows that panel's corner node numbers and copy integer. The next panel's corner node numbers etc. are not read until all the data to be generated for the current panel is complete. Fig. 3.6 shows the arrangement for the temporary storage of data.

3.5.2 Stiffener types.

It is necessary for the stiffness matrix assembly routine to know which type of stiffener applies when adding the stiffener's contribution. This is done by reference to the input listing of their types. The list of types is held in the order in which the assembly routine expects to find them. In the type of structure to be analysed by this package, the same type of stiffener will be seen several times in succession. To avoid unnecessary duplication of input it is only required to add the type number once and follow it by the negative value of the number of times in succession that this value will be repeated. This is then handled by a loop that continues to transfer the same value from the data temporary store to the appropriate location as shown in Fig. 3.12. A second counter must be set during this transfer to take account of the difference in the number of values which have been read in and which have to be stored.

3.5.3 Element mesh.

The surface of each panel is considered to be covered by a coarse rectangular mesh formed by the edges



Note 1

If the current panel's copy integer is set to -j then the first j locations of "Records of data stored" are summed to find the start position in "Stored basic panel data" for suitable data to be used.

Note 2

To economise on space only details of panels which may be used as a copy source are stored. Therefore the Nth panel may not be the previous current panel.

Locations at the end of array Real being used as temporary data storage.

Fig. 3.6

of the panel and all its stiffeners. The space between any pair of adjacent lines of the coarse mesh is divided by equally spaced parallel lines. The number of divisions between coarse grid lines may vary as required and consequently the size of the mesh may differ from one coarse grid to the next but the size of the mesh within any one coarse mesh will be generated as a constant. In the case of a triangular panel the rectangular mesh is parallel to the orthogonal sides.

3.5.4 Element dimensions.

The element dimensions are governed by the number of divisions between stiffeners and the stiffener spacing. This information must be given in the assembly order when preparing the input data. Details of the mesh divisions are stored in a readily accessable location for use as arguments in looping routines for the assembly of the stiffness matrix between stiffeners. The distance between stiffeners is not stored but used to obtain the element sizes which are stored. DIV calls a small subroutine ZERO which is used to relocate blocks of data.

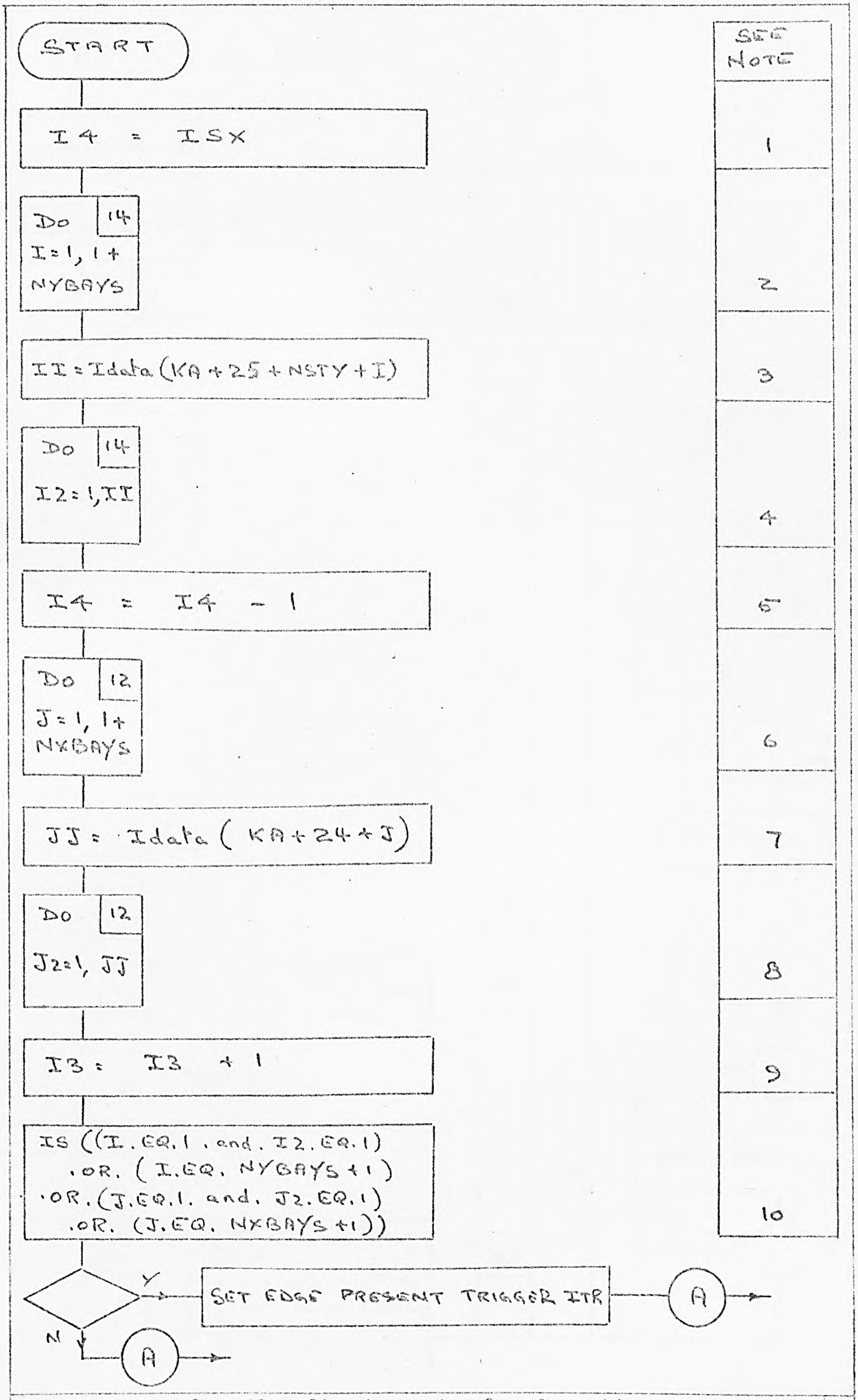
3.6 Subroutine DEGREE.

3.6.1 This routine prepares an index of all the degrees of freedom of the panel for both internal and external without considering any degree of freedom which may be introduced at the edges to take account of transforming selected parts of the local stiffness matrix to global axes. Whilst scanning the whole panel a count is made of every degree of freedom occuring on an edge, this value is returned to FILE as one item of a list of basic pieces of data necessary to set up sizes of segments in the arrays Idata and Real. Since DEGREE scans the whole panel advantage is taken to initiate the bandwidth evaluation which will be completed by ASSEMB. A flow chart of DEGREE is at Fig. 3.7.

3.7 Indexing subroutines.

3.7.1 This package uses the idea of categorising degrees of freedom into either internal or external. Externals are further divided into either active or passive status, with regrading of actives to passives when shown to be necessary by a routine which for now may be considered as a periodic examination of the remaining active freedoms.

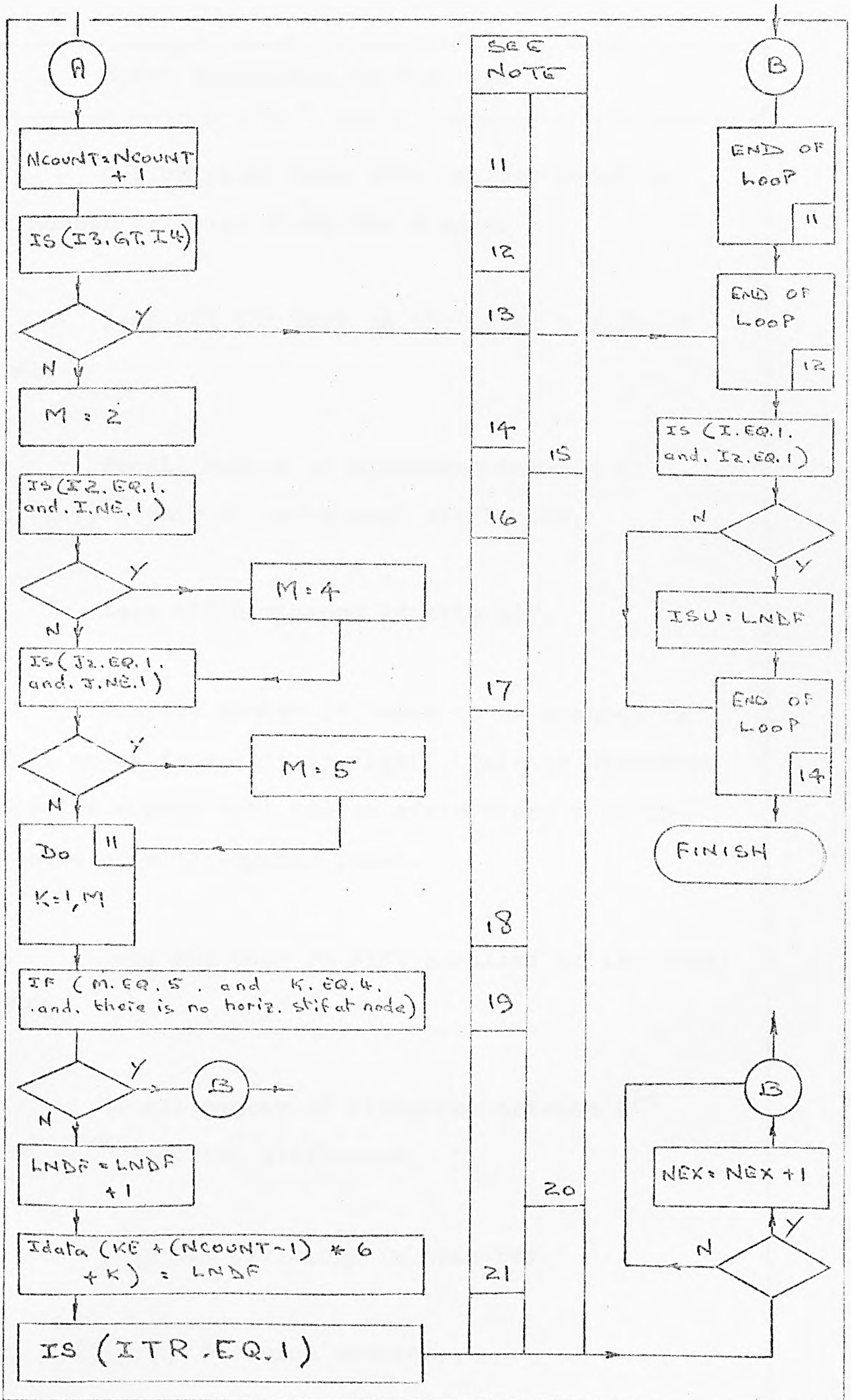
3.7.1.1 An external degree of freedom is one which is physically on one of the edges of the panel or is an internal freedom which has either been manually assigned total restraint or been made a spring ending, in either case this internal freedom is automatically made a psuedo external freedom. Once the assembly routine has identified it as such it is treated thereafter in all respects as an external freedom. Such a psuedo freedom stays as such until after ASSEMB has processed that panel when restrained internal freedoms are made passive. Other psuedo external freedoms remain active until the structure is being reformed when it takes on its true status for the establishment of the displacement pattern.



Flow chart for the first part of subroutine

DEGREE

Fig. 3.7(a)



Flow chart for the remainder of DEGREE.

Fig. 3.7(b)

Notes applicable to Fig. 3.7

1. Set up right hand side trigger equal to the number of nodes along the x axis.

2. Loop all the bays on the local y side of panel.

3. Recall number of divisions between I^{th} and $(I+1)^{th}$ pair of horizontal stiffeners.

4. Loop all divisions in this bay.

5. Correct number of nodes to be scanned in a full sweep from left to right. This is necessary where the number must reduce after every scan in the case of a triangular panel.

6. Loop all bays on side parallel to the local x side.

7. Recall number of divisions between J^{th} and $(J+1)^{th}$ vertical stiffeners.

8. Loop all divisions in this bay.

9. Update division counter.

10. Is the current node set to an edge position?

11. Update node counter.

12. Is this a triangular panel and is the pointer passed the hypoteneuse?

13. If answer to 12 is yes ignore remainder of loop.

14. If answer to 12 is no then set up limit value for number of freedoms that can be expected at next node. In the basic case with no stiffener present there can be only the two plate freedoms.

15. At the end of scanning the first horizontal edge, record number of freedoms so far allocated and store this information for use in subsequent bandwidth calculations.

16. If this is the start of a horizontal bay other than the first then there must be four freedoms to be allocated.

17. If this is the start of a vertical bay other than the first then there must be a vertical stiffener present and thus though there must be four freedoms to be allocated the fourth will go in the fifth location and thus the limiting value must be set to 5.

18. Loop through all locations of freedoms.

19. Skip fourth loop in the case where there is a vertical stiffener present which has caused limit to be set to 5 but there is no horizontal stffener present requiring any value to be put in the fourth location.

20. If this is a node situated on an edge, then update the count of the number of truly external freedoms.

21. Form and allocate the reference number which in this case is related only to the current panel and store it in the KE index.

An internal degree of freedom is one which is neither truely external nor psuedo external.

A passive degree of freedom is an external which has been taken fully into the overall stiffness matrix and has no component left to be added by any subsequent panel.

An active degree of freedom is one being processed or is the next to be processed or one which has been processed in at least one panel's stiffness matrix and which would appear in a list of degrees of freedom of a subsequent panel if such a list existed.

3.7.1.2 Numbering of degrees of freedom.

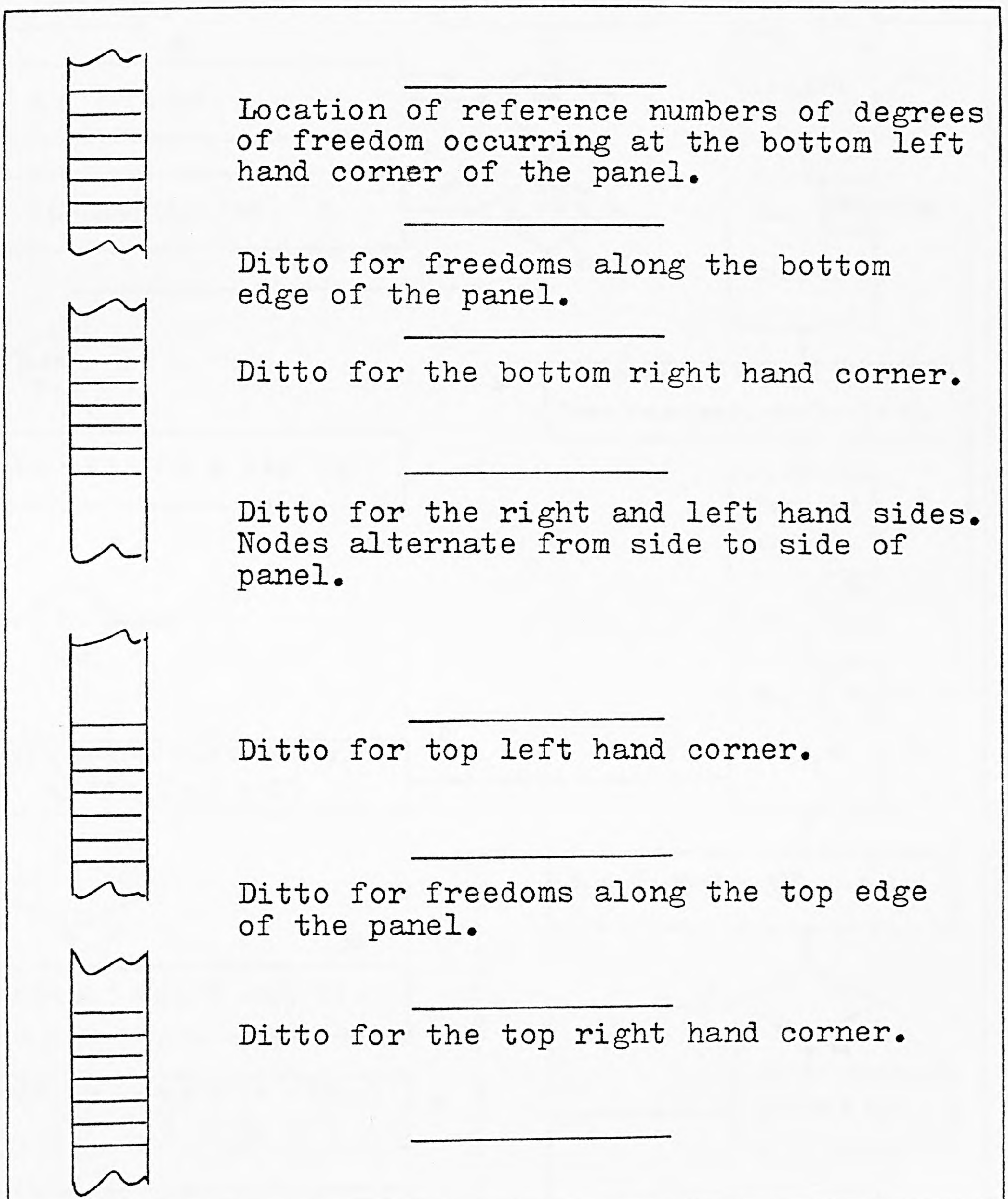
For the program to manipulate the individual rows of the stiffness matrix it must have access to two indices. One containing refence numbers assigned to each degree of freedom being either in an external position of the current panel or allocated external status and as such must not loose their individuality. The second being for all the degrees of freedom on the plate whether internal or external. The generation of these indices is entirely automatic and is prepared by CORNER, EDGE and DEGREE without regard to the status of any psuedo external degrees of freedom. The indices are automatically modified to take account of internally positioned degrees of freedom having external status before assembly commences.

3.7.2 Subroutine CORNER.

3.7.2.1 As implied by the name this subroutine deals with the corners of the current panel and allocates global numbering for the degrees of freedom applying there. Before allocating the numbering it must first ascertain that that the current corner's degrees of freedom have not been already allocated for a previous panel. In the cases where degrees of freedom have already been allocated, the disc file to which that previous panel's index has been written must be found and accessed. The appropriate index values recovered and added to the current index.

3.7.2.2 The refence numbers allocated to external degrees of freedom are stored in a single vector which is a segment of the vector array Idata. These numbers are listed in the order in which the panel is scanned, which is row by row working sequentially along or parallel to the x axis with each row being scanned from the local y axis. Fig. 3.8 shows the form this segment of Idata takes.

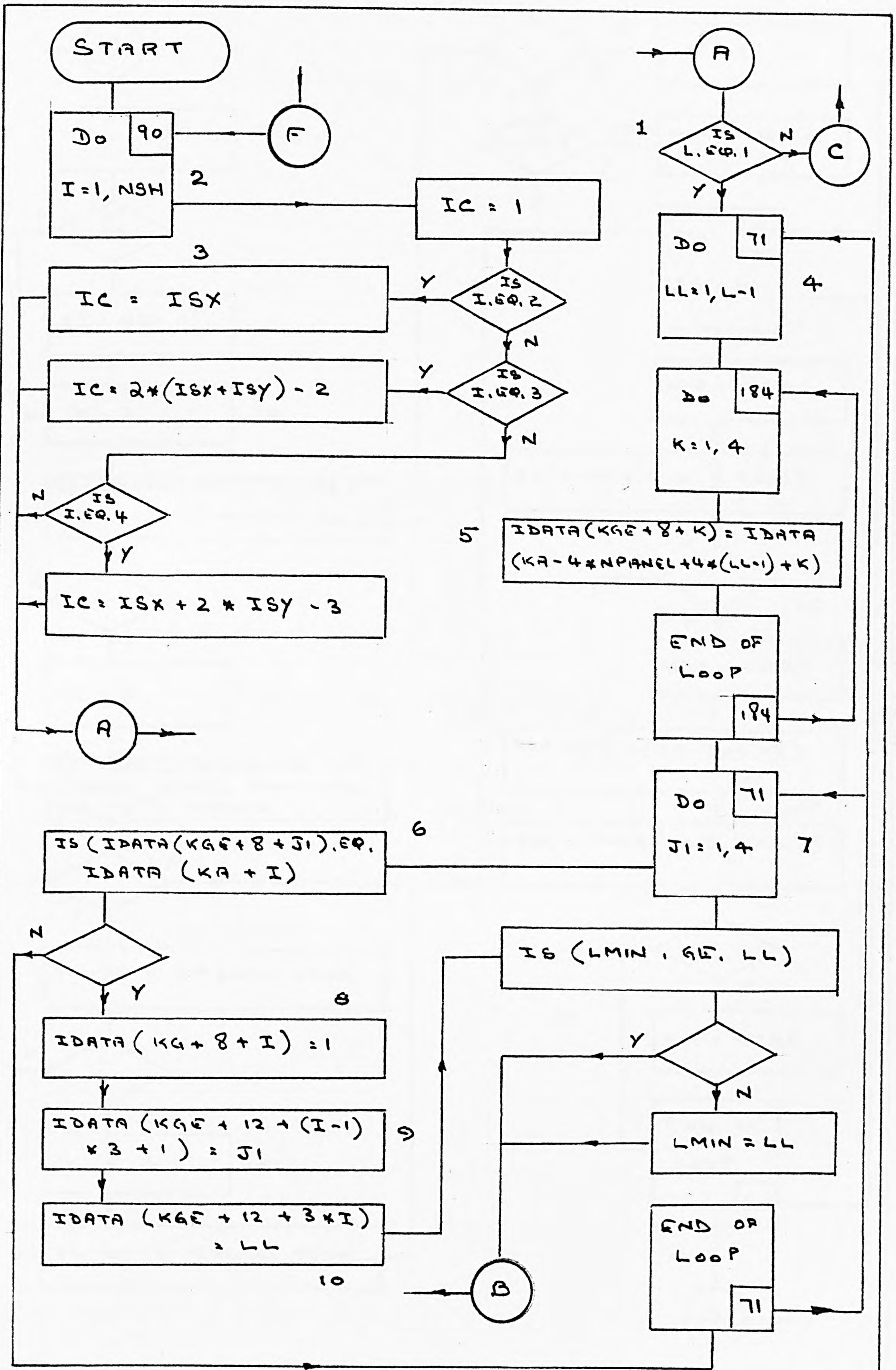
When processing the vertical sides, the technique of scanning to the other side after allocating reference numbers to the current node causes the index to appear shuffled in the section containing the reference numbers from the bottom right hand to the top left hand corners. Allocation in this manner is necessary to be consistent with the assembly routine. Numbering of the sides and corners moves from the origin anti-clockwise round the perimeter of the panel. For triangular panels the hypoteneuse side is treated as the vertical right hand side of a rectangular panel, the allocation routine is then stopped after completing the two vertical sides.



Form of segment of array Idata being used as an index for reference numbers given to perimeter degrees of freedom.

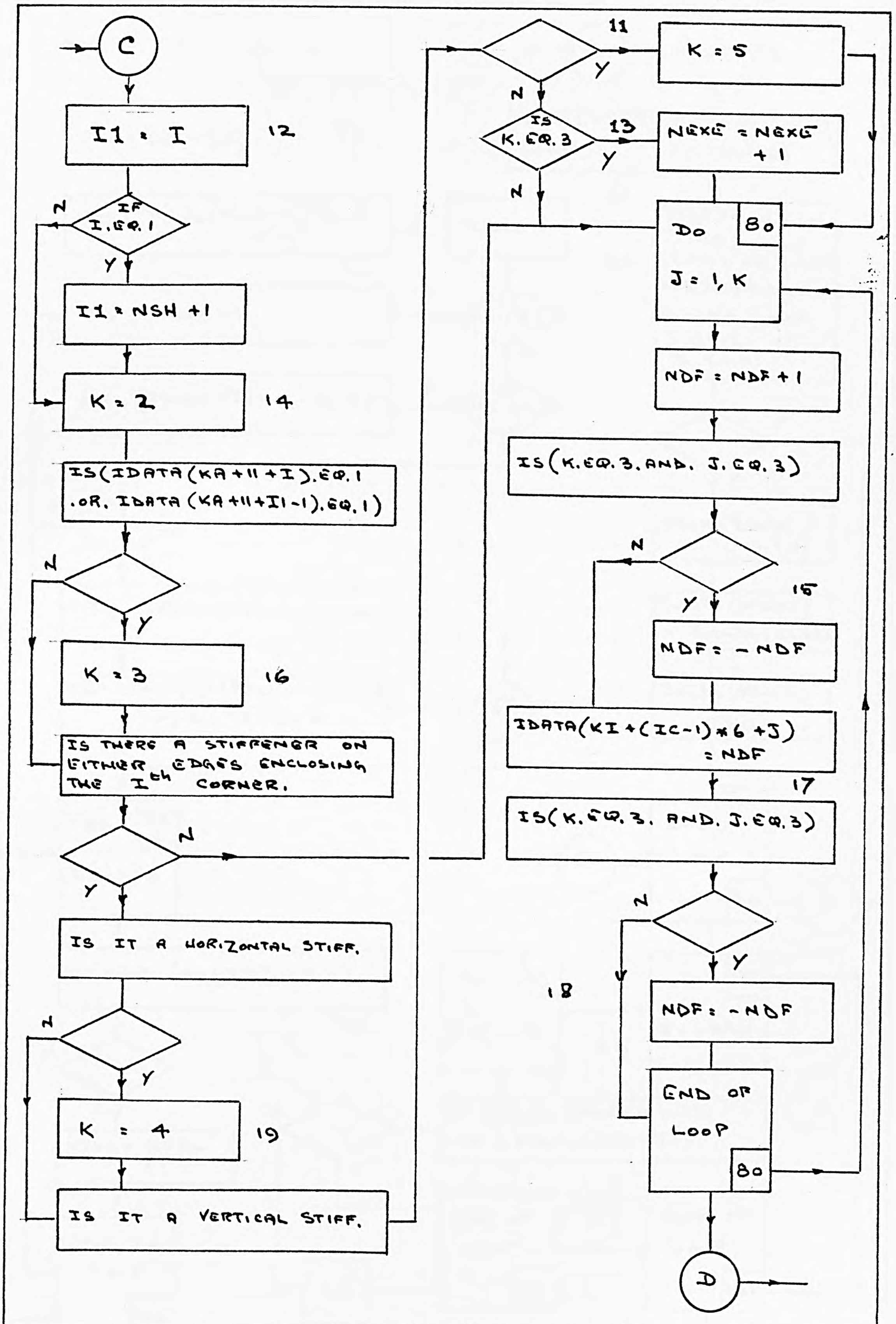
Fig. 3.8

A series of nested loops, see Fig. 3.9 is set up for allocating reference numbers to corners. The outer loop controls the corner being considered. The first inner loop arranges to scan the corner numbers of all previous panels.



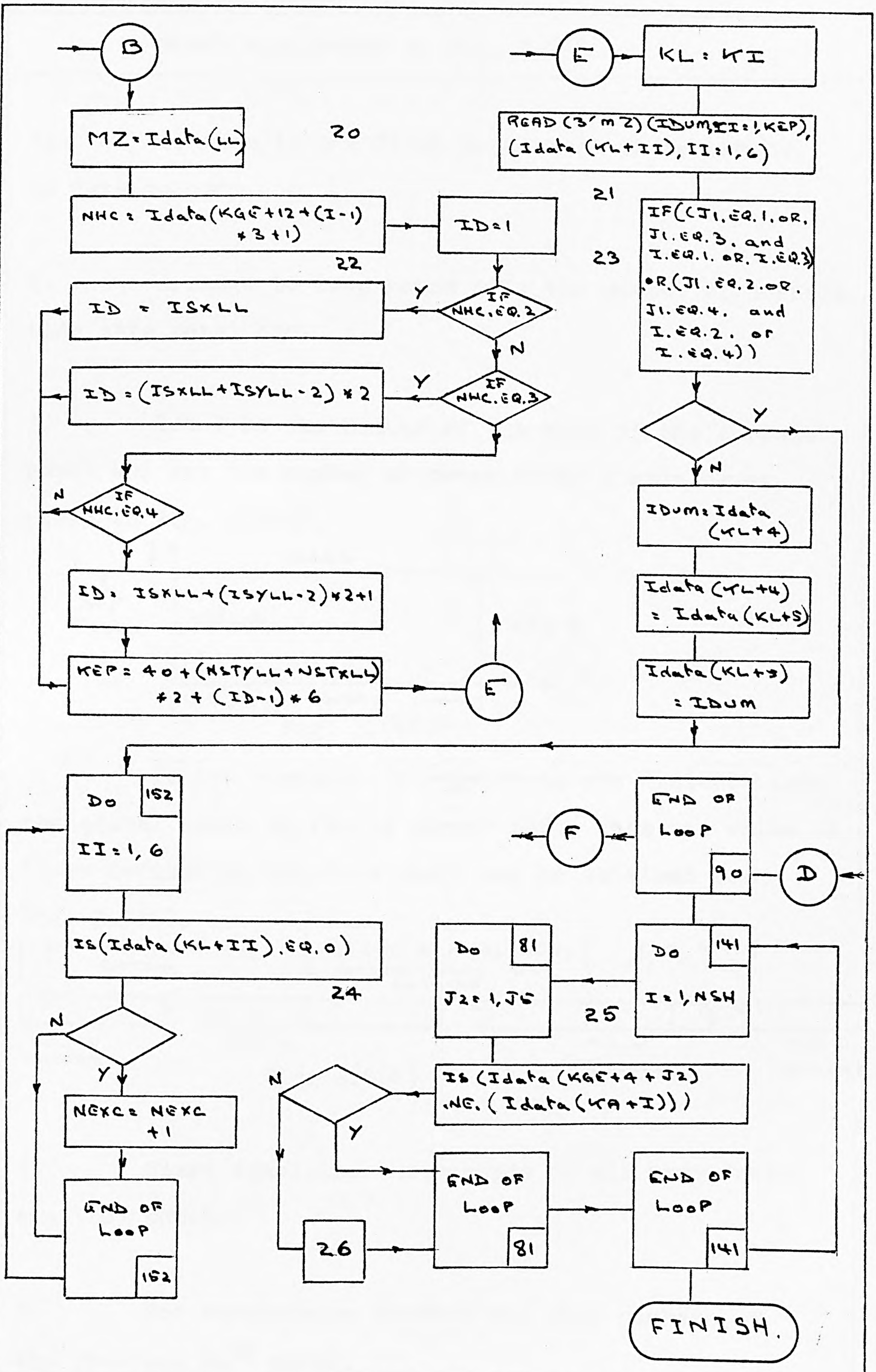
Flow chart showing the method of deriving global degrees of freedom for corner nodes (Part 1).

Fig. 3.9(a)



Flow chart showing the derivation of global degrees of freedom for corner nodes (Part 2).

Fig. 3.9(b)



Flow chart showing the method of deriving global degrees of freedom for corner nodes (Part 3).

Fig. 3.9(c)

Notes applicable to Fig. 3.9

1. If this is the first panel then there can be no data to copy.

2. Arrange to loop round only the number of corners that this panel has.

3. Let I be the number of the side of the current panel and let the number of nodes along a side be as shown in Fig. 3.9(d).

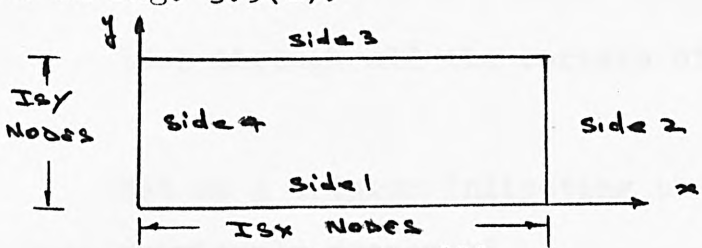


Fig. 3.9(d)

If the variable IC represents the distance into the global index to find a corner node, then the value of IC as defined in the flow chart may be obtained from Fig. 3.9(e).

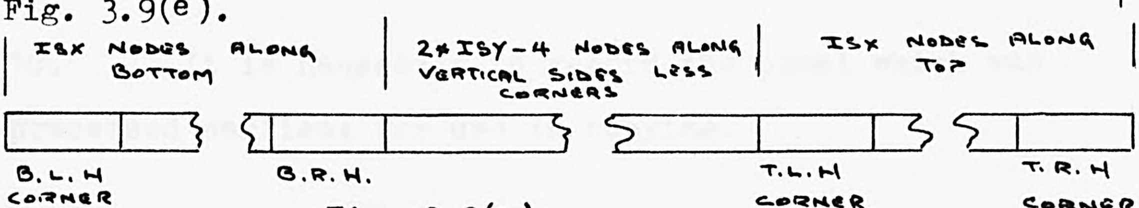
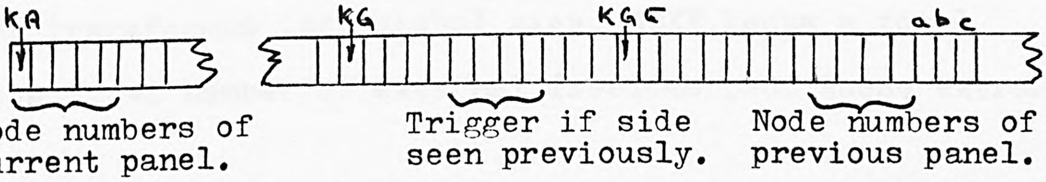


Fig. 3.9(e)

4. Start examining the records of all previously examined panels.

5. For examination recover the node numbers of the previous LL^{th} panel.

6. Compare the I^{th} corner of panel with the $J1^{th}$ corner of the previous panel. The locations of the various variables is shown in Fig. 3.9(f).



a. Number of common corner.

b. Number of second corner if there is a common side.

c. Previous panel number.

Fig. 3.9(f)

7. Loop through all the corners of the LL^{th} panel.

8. Set up a trigger indicating that the I^{th} corner has been previously processed.

9. Set up record giving information on where to find details of corner data which may be copied.

10. It is necessary to record the panel which was processed earliest for use in copying.

11. If there is a stiffener along a vertical side then there will be four locations filled but the fourth will be in the fifth position.

12. Set up a variable to indicate the corner before the I^{th}.

13. In the case where there are no edge stiffeners

at the I^{th} corner but there is an adjacent panel having relative slope, then an extra degree of freedom will need to be introduced to allow for this corner's freedoms being transformed into global axes. NEXE keeps a total of the known number of external freedoms plus these extras.

14. Set up limiting value of loop to insert degrees of freedom for a plate element with no stiffener attached.

15. When an extra freedom is necessary as described in 13 above, it is put in the index as a negative to trigger subsequent procedures.

16. Loop limit must be 3 if any of the panels meeting at the I^{th} corner has a relative slope. This will be overwritten if there is a stiffener on either of the enclosing sides.

17. Arrangements are made to skip if K has been set to 5 when J becomes 4 with no applicable freedom.

18. Put the negative freedom value back to positive to prevent it upsetting the freedom counting.

19. If there is a stiffener along the horizontal side, the first four locations will be filled, if there is a stiffener on both edges then all five locations will be filled.

20. Recover disc record number for the LL^{th} pane. At this time data is also recovered from array Idata for

information to assist in the search through that disc's record for the LL^{th} panel's corner reference numbers.

21. Recover reference numbers from disc.

22. Form variable for number of nodes on record to be scanned to find information.

23. When copying reference numbers, if two panels are not suitably aligned relative to each other the fourth degree of freedom of one panel will be the fifth of the other and vice versa.

24. Sum the freedoms just read from disc with the total of like freedoms already counted. The final total will be used to correct the number of freedoms existing and prevent this being counted now as well as at the time of processing the LL^{th} panel.

25. Before starting subroutine EDGE prepare records of the availability of a previously processed side which is suitable and available for copying from. This is done now because the current searches are through corners and if the corners at both ends of the current edge are found during the search then there must be a processed edge to copy.

26. Compare corner numbers, if a positive result is obtained go on and check the corners either side of it.

End of notes for Fig. 3.9

It then checks that reference number against the reference number of the current corner being considered. There are now two possibilities, either a common node number is found, in which case all the freedom values at that node can be recovered and put into the current record or the node number has never been seen before in which case the next available reference numbers must be allocated.

3.7.2.3 Action to be taken if a previous panel is found to contain a corner whose reference number is the same as that of the current corner.

The format for the storage of global reference numbers of degrees of freedom though standard throughout the program, is different in absolute values. Reference to Fig. 3.8 shows that if the number of nodes along the local axes of a panel are known, then the location of any corner node's freedom numbers to be recovered can be found. To avoid time wasting calls to disc whilst data is being generated, a small amount of key information for all panels thus far processed is held in core and is then immediately available to give a value for how much of the disc record needs to be ignored to reach the numbers to be copied. The numbers are copied directly into their correct positions in the core.

If the corner number is found , then there exists the possibility that a side of a previous panel having this corner as one end has had reference numbers allocated for all the degrees of freedom along its length. It is not necessary that the panel found in this way is the same panel as that which had the first common corner discovered in the first checking scan. Consider Fig. 3.10 when panel 3 is being processed a scan of the details appertaining to the previously processed panel 1 will find the third corner is available to be copied for the first corner of panel 3. This corner is also the second corner of panel 2 and is one end of a side which is common with the current panel. Therefore though one panel may produce a suitable corner for copying a search must be made through the other panels to establish

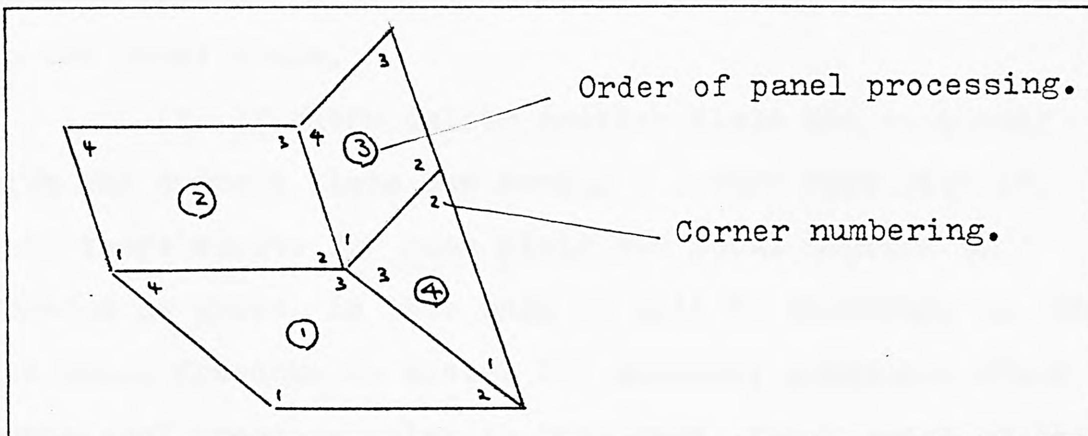


Fig. 3.10

whether a more suitable panel exists. Though this part is concerned with the sides and not with freedom numbers of corners, it is dealt with at this stage because it is more expeditious here since the criteria are dependent upon corner nodes. A trigger of three integers is set up in a twelve location section of the Idata array, storing for each side of the current panel the relevant trigger. These integers being the number of the panel containing the common side, which side is the common side of the previous panel and which side is the common side of the current panel.

3.7.2.4 Action to be taken if no reference can be found to the current corner as having been previously processed.

At any corner node there exist four possibilities for the number of freedoms existing there. The possibilities depend upon the state of the plate edges' which contain that corner.

(a) Both edges are free of stiffeners along their length and if there are other plates sharing either edge as a common edge then all of these plates are co-planar. In this case there can be only two freedoms at that corner in the local plane.

(b) If there exists another plate not co-planar with the current plate but having a common edge with it, then there exists for each plate two local degrees of freedom as above. In this case it will be necessary to change the local freedoms to global for assembly purposes. Since no rotational freedoms exist in this case, there exist at this node three translational freedoms.

(c) If a stiffener exists along one edge then at the corner there will exist the two translation freedoms from the plate plus one rotational and two translational degrees of freedom from the stiffener. One of these translational freedoms will be common to one of the plate's freedoms. Therefore when a stiffener exists along one edge there will be four degrees of freedom. The orientation of the stiffener in relation to the local axes of the plate will decide whether the rotational freedom reference number is stored in the fourth or fifth location of the set of locations containing the reference numbers of all the degrees of freedom of this node. A rotational contained in the plane at right angles to the first translational freedom is stored in the fourth location and a rotational contained in the plane at right angles to the second translational freedom is stored in the fifth position. Any configuration which introduces a sixth freedom at a corner is not allowed.

3.7.3 Subroutine EDGE.

As CORNER allocates global reference numbers to corner degrees of freedom so this subroutine does the same for edges.

3.7.3.1 Action to be taken if the current side is not a side which is common to both the current panel and any panel which has been already processed.

This is very similar to the treatment of a corner not having been seen before. The action applied to a corner is repeated for the whole of the side but not including the nodes at the ends. When considering extra freedoms due to the slope of adjacent panels only one panel need be considered for each side of the current panel as opposed to two for each corner.

As explained, the six locations for each node along the first and third sides are stored consecutively, but those on the second and fourth sides are intermingled switching alternately from side to side, starting with the fourth side.

3.7.3.2 Action to be taken if the current side has already been allocated reference numbers for freedoms along its length.

A complication occurs with EDGE which cannot apply to CORNER, because CORNER deals with only one node at a time. This complication is that it may not be possible to copy directly the reference numbers which apply to the current edge and which are available in the record of a previous panel's generated data. For any panel the reference numbers of the degrees of freedom of the first and second sides are stored in the direction defined by the order in which the corner numbers were read. In the case of the third and fourth edges the order of storing is the direction opposite to the order of reading their corner numbers. Based upon this rule the table shown in Fig. 3.11 has been prepared.

The recovery and writing to their new locations of these common freedom numbers is further complicated since for any panel the second and fourth edges have the reference numbers of freedoms stored alternately whilst the other sides are in consecutive order. So that to recover data of a fourth side six locations must be ignored after reading every six, whilst for second side data the six must be read after ignoring six.

Similarly when writing to store details of the second and fourth sides, six locations must not be written into, after or before every active six. In the above mentioned reversing of the reading of node order, with the exception of some fourth and fifth order freedoms the reversing does not apply to the individual reference numbers allocated to a node. The details of each node are read from

Previously processed panel's side number to be copied.	Number of side of current panel recovering data.	Comparison of order of corner numbers of the two panels.	Direction of reading data.
1 or 2	1 or 2	Same	As written
		Opposite	Reversed
	3 or 4	Same	Reversed
		Opposite	As written
3 or 4	1 or 2	Same	Reversed
		Opposite	As written
	3 or 4	Same	As written
		Opposite	Reversed

Table indicating direction for reading reference numbers of degrees of freedom of previously processed sides of panels.

Fig. 3. 11

right to left even if the nodes are being read from left to right. In the case of fourth and fifth order freedoms, a fourth order freedom is a rotation about an axis parallel the local y axis. When copied it may be about an axis parallel the local x axis and thus becomes a fifth order freedom. Similarly a fifth order may become a fourth order freedom.

In addition to the normal case where stiffeners run from panel to panel, this package also allows stiffeners to exist on one side of a common edge but not to be carried over the edge onto the next panel. In this case allowance must be made to suppress freedom numbers where the current panel does not contain any stiffeners which match the details recovered. Conversely extra freedoms must be generated to supplement recovered data where there are no stiffener details to apply to existing stiffeners.

3.7.3.3 In the case of triangular panels the program treats them as rectangular panels and ignores the third side. A rotational freedom for a stiffener crossing an hypoteneuse side may introduce a component about the global z axis. This is a configuration which happens infrequently and to save core space is not allowed in the package. Where an hypoteneuse is a common edge, reference numbers are recovered from stored data of the previous panel which shares this edge as already explained. But in this case the rotational freedom of the second panel is given a new number. When the stiffness matrix is assembled the rotational freedoms at that node due to the two stiffeners

DETAILS OF DISC RECORDS.	2*NPANEL+2
CORNER NODE NUMBERS, SUMMED FOR EACH CORNER	NCORN
TEMPORARY STORAGE	10
INTERNAL AND EXTERNAL NUMBERS OF FREEDOM	2*NPANEL+1
STIFFENER DETAILS	4*NUMBER OF STIFFENERS +1
READY USE STORE OF KEY VARIABLES.	8*NPANEL
CORNER NODE NUMBERS	4
NUMBER OF OTHER PANELS JOINING AT THAT CORNER	4
NSTY	
NSTX	
ISUM	
SLOPE OF ADJACENT PANELS	4
NSP	
NEX	
NIF	
NE	
NDF	
NXBAYS	
NYBAYS	
NT	
ISY	
NUMBER OF SPACES BETWEEN STIFFENERS PARALLEL Y AXIS.	NSTY +1
DITTO FOR X AXIS	NSTX +1
TRIGGERS FOR TRANS. TO ROTATE ROTATIONAL FREEDOMS ON NON COMPATIBLE EDGES.	4
TRIGGER FOR PRESENCE OF STIFFENER ON EDGE OF PANEL	4
NIDISP	
NILOAD	
COMMON SIDE DETAILS.	4
LIST OF STIFFENER TYPES	NSTY + NSTX
EXTERNAL FREEDOMS' GLOBAL INDEX	(ISX+ISY-2)*12
INDEX OF ALL FREEDOMS OF PANEL.	ISX*ISY*12
THICKNESS REPEAT COUNTS	NT
LOCAL REFERENCE NUMBERS OF FREEDOMS WITH APPLIED LOADS.	NILOAD
LOCAL AND GLOBAL FREEDOMS HAVING 100 % RESTRAINT.	2*NIDISP
SPRING DETAILS, FREEDOM NUMBER, SPRING NUMBER and TYPE.	3*NSP

The location of integer data in array Idata and space occupied by each group at the end of subroutine FILE.

Fig. 3.12

are not added, because they are now identified by different numbers. The other freedoms present at that node are added as normal.

In the case of two rectantangular panels having a common edge but so defined that their rotational freedoms at a stiffener crossing this edge are not in the same rotational direction, then their stiffness coefficients are not suitable for direct addition as is required for this type of freedom. This situation is catered for by a trigger being set if when the recovered freedom reference number for the common edge when the second panel is being processed are required to be read in their reverse order. This trigger then causes PARTIT to be made ready when this freedom is met.

When one or more triangular panels share a common edge with a rectangular panel, the rectangular panel must be assembled before any of the triangular panels. If this is not done the stiffeners of the rectangular panel will copy the rotational freedom reference number of the triangular panel's stiffener, thus causing an addition of incompatible stiffness terms. If the triangular panels are all copiers then the above mentioned trigger will be set.

4. The use of Finite Element Analysis.

4.1 Application to ship structures.

4.1.1 Analytical methods have been applied to ship structures for many years. In 1874 John (20) enunciated the principals of the longitudinal wave bending moment component. The first recorded attempt to estimate the transverse strength of a ship was by Read and Jenkins (21) in 1882. With the the advent of the high speed computer it is now possible to predict stresses and deflections in highly complex structures such as ships, by the use of the finite element method, to a degree of accuracy undreamt of by those early analysts.

4.1.2 The pioneering efforts (22),(23),(24) behind this method began in the early 1950's primarily in the aircraft industry. The designers of ship structures have been slow to use this tool. The first published paper using the technique for ship design was published in 1960 (25). Hughes (26) has 220 references in his paper and none of these refers to ship design or ship structures. This lack of representation is probably due to the small size of the industry. Finite element packages tailored to the marine environment such as Sesam (27) developed by Det norske Veritas and LR Safe (28) developed by Lloyd's Register of Shipping from Nastran (29) are now available.

4.2 Force - displacement relationship.

4.2.1 It will now be shown that there exists a relationship between the forces applied to a structure and the resulting displacements which can be written as

$$P = K * U$$

where K is known as the structural stiffness matrix of the structure and P and U are the vectors of the applied loading and resulting displacements.

To determine the value of this structural stiffness matrix, it is necessary to idealize the structure into an assembly of elements whose whole response will as far as possible represent the structure's response to the same loadings. The degree of realism in the response of the idealized structure will depend on the accuracy of the assembled stiffness matrix. A fundamental step in the application of the matrix displacement method is the determination of the stiffness characteristics of the structural elements, which are summed to give the whole structure's stiffness matrix. In general the relationship between loads and displacements for the whole structure is written in upper case letters and for structural elements in lower case.

4.2.2 Deformation of structures.

Consider a system of concentrated forces as shown in Fig.4.1 acting on elastic body, which is equilibrium under their action. Suppose a subsequent distortion of the body, due to the application of a second force system, causes deflections u_1, u_2, ... u_n to occur at the load points of the initial set of forces and in the direction of these forces. Concentrated forces may include moments. The work done, V by the initial set of forces during the subsequent deformations of the structure is given by

$$V = p_1u_1 + p_2u_2 + \dots + p_nu_n \qquad \dots(4.1)$$

It is convenient to represent the loads and deflections as column matrices

$$p_v = \begin{bmatrix} p_1 \\ p_2 \\ \cdot \\ \cdot \\ \cdot \\ p_n \end{bmatrix} \qquad u_a = \begin{bmatrix} u_1 \\ u_2 \\ \cdot \\ \cdot \\ \cdot \\ u_n \end{bmatrix}$$

This allows the work done V to be written

$$V = p_v^T\, u_a \qquad \dots(4.2)$$

The subscipts v and a will be used subsequently to denote effects caused by the initial set forces and the subsequent set of loads respectively.

An alternative method of evaluating the work done V is to divide the body into a large number of elemental parallelepipeds having faces parallel to some chosen rectangular system of axes. The load system p_v will cause a distribution of internal load throughout the structure and the stresses caused by this load system on a typical elemental volume of the structure are as shown over the page.

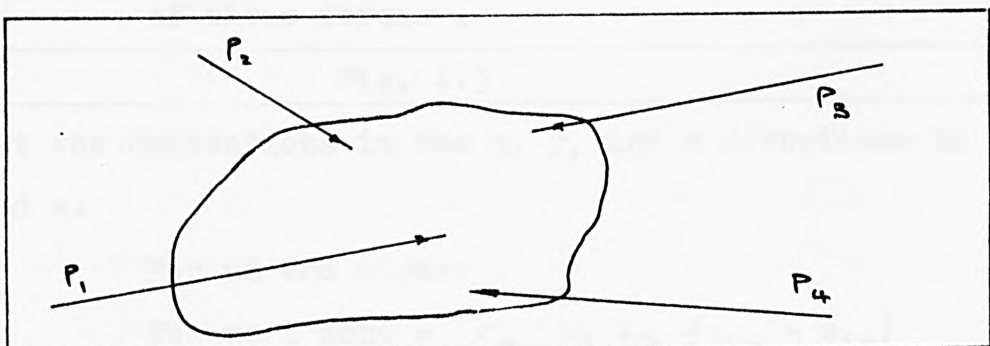


Fig. 4.1

It is assumed now that when the second system of forces is applied to the structure the deformation which

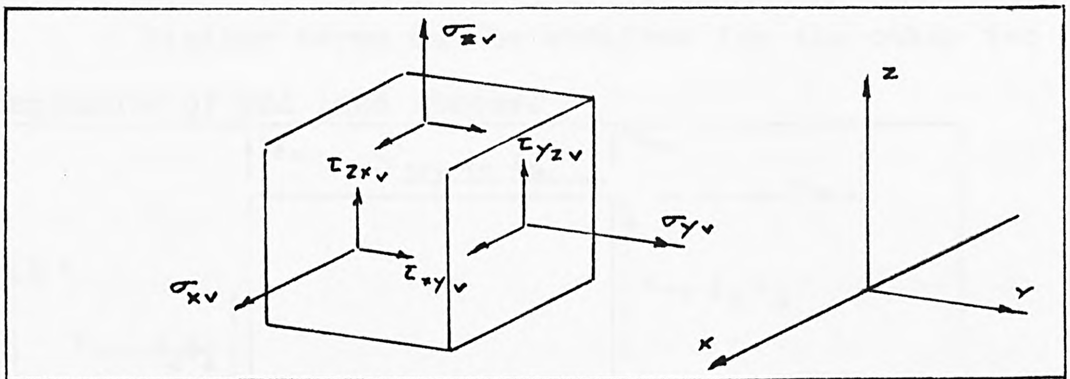
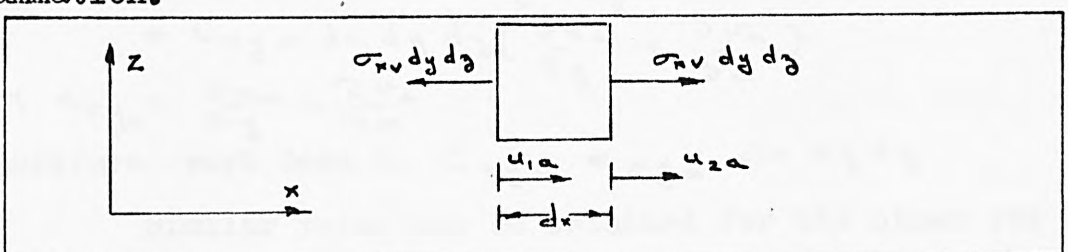


Fig. 4.2

occurs is small and that the pattern of internal forces which balance p_v is not changed by this distortion.

For a typical parallelepiped the work done by the internal forces balancing p_v during the subsequent deformation of the structure may now be evaluated. This will be done separately for each of the six components of stress acting on the volume and the final result then achieved by summation.



Forces on the faces of a typical element from σ_{xv} and deflection in the x direction of those forces.

Fig. 4.3

Let the deflections in the x, y, and z directions be u, v, and w.

For an end stress

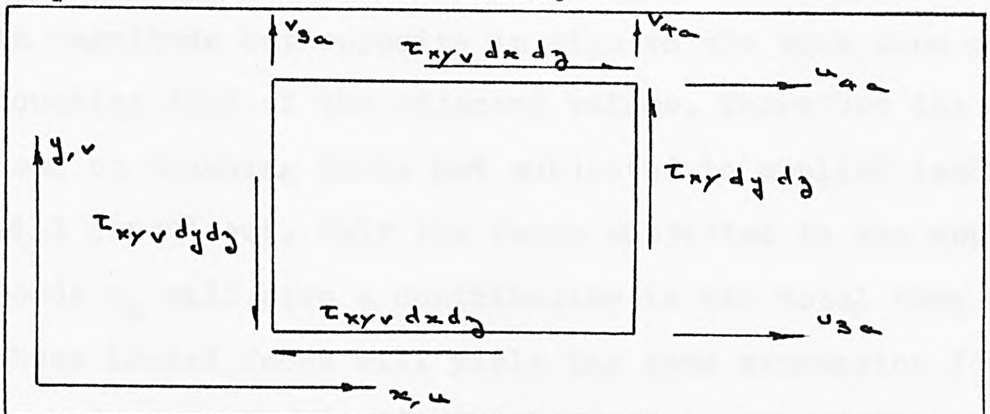
The work done = $\sigma_{x_v}\, dy\, dz\, (u_{2a} - u_{1a})$

Since $e_{xa} = \partial u/\partial x = (u_{2a} - u_{1a})/\,dx$

therefore work done $= \sigma_{xv}\, dy\, dz\, (e_{xa}\, dx)$

$= \sigma_{xv}\, e_{xa}\, dx\, dy\, dz$

Similar terms may be obtained for the other two components of end load stress.



Forces on the faces of a typical element from shear stress τ_{xyv} and tangential displacement of the sides.

Fig. 4.4

For a shear stress, the work done

$$= \tau_{xyv}\, dx\, dz\, (u_{4a} - u_{3a}) + \tau_{xyv}\, dy\, dz\, (v_{4a} - v_{3a})$$

$$= \tau_{xyv}\, dx\, dy\, dz \left\{ \frac{(u_{4a} - u_{3a})}{dy} + \frac{(v_{4a} - v_{3a})}{dx} \right\}$$

$$= \tau_{xyv}\, dx\, dy\, dz \left(\frac{\partial u_a}{\partial y} + \frac{\partial v_a}{\partial x} \right)$$

but $e_{xya} = \dfrac{\partial u_a}{\partial y} + \dfrac{\partial v_a}{\partial x}$

therefore work done $= \tau_{xyv}\, e_{xya}\, dx\, dy\, dz$

Similar terms may be obtained for the other two components of shear stress.

The total work done dV on the volume by the internal forces balancing p_v when the subsequent deformations occur is given by

$$dV = (\sigma_{xv} e_{xa} + \sigma_{yv} e_{ya} + \sigma_{zv} e_{za} + \tau_{xyv} e_{xya} + \tau_{yzv} e_{yza} + \tau_{zxv} e_{zxa})\, dx\, dy\, dz$$

The result of adding together the work done on all such elemental volumes will now be considered. From equilibrium considerations, parallelepipeds which are adjacent to each other will have forces on their touching faces which are equal in magnitude but opposite in direction. For a compatible deformation of the structure the touching faces must remain in contact and have equal deflections. The work done by the forces on one face will therefore be equal in magnitude but opposite in sign to the work done on the touching face of the adjacent volume. Therefore the work done on touching faces not subjected to applied loading will cancel out. Only the faces subjected to the applied loads p_v will give a contribution to the total work done. These loaded faces will yield the same expression for the work done as Eq. (4.2). Hence providing the deflection pattern is compatible then the total work done obtained by summing the work done on each of the elemntal volumes will be the same as the work done obtained by multiplying each of the applied forces by the appropriate displacement.

$$V = p_v^T u_a = \iiint (\sigma_{xv} e_{xa} + \sigma_{yv} e_{ya} + \sigma_{zv} e_{za} + \tau_{xyv} e_{xya} + \tau_{yzv} e_{yza} + \tau_{zx} e_{zxa}) \, dx.dy.dz$$

Using column matrix representation again this equation may be written as

$$p_v^T u_a = \iiint \sigma_v^T e_a \, dx \, dy \, dz \qquad ...(4.3)$$

If a single unit virtual load is applied to a body which is fixed to rigid reaction points then Eq. (4.3) becomes

$$1 \, u_a = \iiint \sigma_v^T e_a \, dx \, dy \, dz \qquad ...(4.4)$$

This formula may be used to derive the deflections at selected positions on a body.

4.3 Stress - strain equations.

4.3.1 The total strain at each part of a cooled body also subjected to physical external loading can be thought of as consisting of two parts. The first part being the elastic strain which is required to maintain the displacement continuity of the body and the second part the thermal strain due to the uniform thermal contraction. Hence

Total strain = Elastic strain + Thermal strain

In this work the structure is considered to be subjected to plane stress. Using Hooke's law for linear isothermal elasticity, this strain relationship may be expressed in the following matrix form

$$\begin{bmatrix} e_x \\ e_y \\ e_{xy} \end{bmatrix} = \frac{1}{E}\begin{bmatrix} 1 & -\nu & 0 \\ -\nu & 1 & 0 \\ 0 & 0 & 2(1+\nu) \end{bmatrix}\begin{bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{bmatrix} + \begin{bmatrix} \alpha T \\ \alpha T \\ 0 \end{bmatrix} \quad \ldots(4.4)$$

This may be rewritten as

$$\begin{bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{bmatrix} = \frac{E}{(1-\nu^2)}\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{(1-\nu)}{2} \end{bmatrix}\begin{bmatrix} e_x \\ e_y \\ e_{xy} \end{bmatrix} + \frac{E\alpha T}{(1-\nu)}\begin{bmatrix} -1 \\ -1 \\ 0 \end{bmatrix} \quad (4.5)$$

Which may be written as

$$\sigma = [D]\, e + \alpha T\, [D_T] \quad \ldots(4.6)$$

4.3.2 Due to the linear relationship considered to apply in this analysis, the total strains e are directly related to displacements.

Therefore $e = [b]\, u$ ...(4.7)

where $[b]$ represents a matrix of the exact strains due to unit displacements. In this chapter only the elastic strain components of the whole will be considered.

So Eq. (4.6) reduces to $\sigma = [D]\, e$ ...(4.8)

Substituting Eq. (4.7) into Eq. (4.8) gives

$$\sigma = [D][b]\, u$$

Transposing

$$\sigma^T = u^T [b]^T [D]^T$$

Substituting Eq. (4.7) and Eq. (4.8) into Eq. (4.3) gives

$$p_v^T u_a = \iiint u_v^T [b]_v^T [D]_v^T [b]_a \, dx\, dy\, dz$$

Transposing Eq. (4.1)

$$p_v^T = u_v^T [k]^T$$

Post multiplying both sides by u_a

$$p_v^T u_a = u_v^T [k]^T u_a$$

but

$$k^T = k$$

therefore

$$u_v^T k\, u_a = \iiint u_v^T [b]^T [D]^T [b]\, u_a \, dx\, dy\, dz$$

Comparing both sides of this equation leads to a value for the stiffness matrix k

$$k = \iiint [b]^T [D]^T [b] \, dx\, dy\, dz$$

and if the thickness is uniform then

$$k = t \iint [b]^T [D]^T [b] \, dx\, dy \qquad \ldots(4.9)$$

4.4 Rectangular plate element stiffness matrix.

4.4.1 In this work only simple elements are used. More comlex elements may be used if modifications discussed in chapter 11 are implemented in the data generating and assembly routines. Two types of rectangular plate elements are available in the package. The first type is the usual element having linearly varying boundary displacements. It will be shown later in this chapter that this type makes a poor showing in situations of non linear strain. The second type is derived from the displacements occuring after a realistic distortion pattern has been applied. One type of triangular plate element is available, which having a linearly varying displacement leads to a constant stress value within the element. One type of beam element is used, once again a very simple idealization but quite suitable for problems where all the stiffeners have a uniform cross section for their length on their panel and resistance to bending in the plane parallel to the plating may usually be ignored. In the idealization used no stiffener can exist without an associated plate, though this may be sufficiently thin as to have no effect on the results. Similarly dummy stiffeners may be used having no strength. The stiffness matrices derived in this chapter are not used in that form they are re-expressed in chapter 6 in a form compatible with row by row assembly.

4.4.2 Derivation of the first type of plate element.

Let the displacements of a point (x,y) be respectively u and v onto the local x and y axes. Let u and v be equated by polynomials in terms of N, the shape function such that

$$u = N_1u_1 + N_2u_2 + N_3u_3 + N_4u_4 \qquad \ldots(4.10)$$

and

$$v = N_1v_1 + N_2v_2 + N_3v_3 + N_4v_4 \qquad \ldots(4.11)$$

where u_i and v_i are the x and y displacements of node i.

If N_i is defined as $(f(x) + a)(f(y) + b)/4ab$ in which the functions f are such that N_i is 1 when the co-ordinates (x,y) coincide with the ith corner and all other N_is become zero.

Fig. 4.5 shows the order of numbering the nodes for this chapter. The order for data input purposes is anti clockwise.

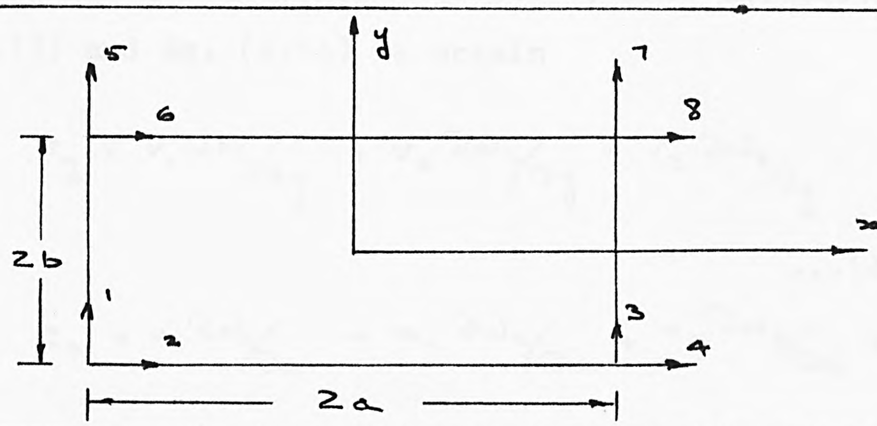


Plate element dimensions, node numbering and order of degrees of freedom.

Fig. 4.5

Consider the case where i = 4, substituting f(x) = a and f(y) = b we get $N_4 = 1$ and the other N values are zero as required. But if we consider any other corner say i = 2 then f(x) = a and f(y) = -b, now $N_2 = 0$ which is not the value needed. This anomoly can be avoided by using the following table applicable for the corner numbering used here.

i	f(x)	f(y)
1	-x	-y
2	x	-y
3	-x	y
4	x	y

Table for use in forming shape functions.

Fig. 4.6

It can be seen this this gives a linear distribution of displacements between nodes.

By definition

$$e_y = \partial v / \partial y \qquad \ldots(4.12)$$

$$e_x = \partial u / \partial x \qquad \ldots(4.13)$$

$$e_{xy} = \partial u/\partial y + \partial v/\partial x \qquad \ldots(4.14)$$

Substituting Eq. (4.10) and Eq. (4.11) into Eq. (4.12), Eq. (4.13) and Eq. (4.14) we obtain

$$e_y = v_1 \partial N_1/\partial y + v_2 \partial N_2/\partial y + v_3 \partial N_3/\partial y + v_4 \partial N_4/\partial y \qquad \ldots(4.15)$$

$$e_x = u_1 \partial N_1/\partial x + u_2 \partial N_2/\partial x + u_3 \partial N_3/\partial x + u_4 \partial N_4/\partial x \qquad \ldots(4.16)$$

$$e_{xy} = u_1 \partial N_1/\partial y + u_2 \partial N_2/\partial y + u_3 \partial N_3/\partial y + u_4 \partial N_4/\partial y + v_1 \partial N_1/\partial x + v_2 \partial N_2/\partial x + v_3 \partial N_3/\partial x + v_4 \partial N_4/\partial x \qquad \ldots(4.17)$$

Using the table at Fig. 4.6 gives

$$N_1 = (a-x)(b-y)/4ab$$
$$N_2 = (a+x)(b-y)/4ab$$
$$N_3 = (a-x)(b+y)/4ab$$
$$N_4 = (a+x)(b+y)/4ab$$

Differentiating with respect to x and y

$$\partial N_1/\partial x = -(b-y)/4ab \qquad \partial N_1/\partial y = -(a-x)/4ab$$
$$\partial N_2/\partial x = (b-y)/4ab \qquad \partial N_2/\partial y = -(a+x)/4ab$$
$$\partial N_3/\partial x = -(b+y)/4ab \qquad \partial N_3/\partial y = (a-x)/4ab$$
$$\partial N_4/\partial x = (b+y)/4ab \qquad \partial N_4/\partial y = (a+x)/4ab$$

Hence

$$e_y = -v_1(a-x)/4ab - v_2(a+x)/4ab + v_3(a-x)/4ab + v_4(a+x)/4ab \qquad \ldots(4.18)$$

$$e_x = -u_1 (b-y)/4ab + u_2 (b-y)/4ab - u_3 (b+y)/4ab + u_4 (b+y)/4ab$$

...(4.19)

$$\begin{aligned} e_{xy} = & -u_1 (a-x)/4ab - u_2 (a+x)/4ab \\ & + u_3 (a-x)/4ab + u_4 (a+x)/4ab \\ & - v_1 (b-y)/4ab + v_2 (b-y)/4ab \\ & - v_3 (b+y)/4ab + v_4 (b+y)/4ab \end{aligned}$$

...(4.20)

Putting this in matrix notation

$$\begin{bmatrix} e_y \\ e_x \\ e_{xy} \end{bmatrix} = \frac{1}{4ab} \begin{bmatrix} & b & \end{bmatrix} \begin{bmatrix} v_1 \\ u_1 \\ v_2 \\ u_2 \\ v_3 \\ u_3 \\ v_4 \\ u_4 \end{bmatrix}$$

Where $\begin{bmatrix} b \end{bmatrix}$ is

$$\begin{bmatrix} -(a-x) & 0 & -(a+x) & 0 & (a-x) & 0 & (a+x) & 0 \\ 0 & -(b-y) & 0 & (b-y) & 0 & -(b+y) & 0 & (b+y) \\ -(b-y) & -(a-x) & (b-y) & -(a+x) & -(b+y) & (a-x) & (b+y) & (a+x) \end{bmatrix}$$

Which is of the form

$$e = \begin{bmatrix} b \end{bmatrix} u$$

...(4.21)

Multiplying $\left[b\right]^T \left[D\right]$ gives

$$\frac{E}{(4ab)^2(1-\nu^2)}\begin{bmatrix} -(a-x) & -\nu(a-x) & -(1-\nu)(b-y)/2 \\ -\nu(b-y) & -(b-y) & -(1-\nu)(a-x)/2 \\ -(a+x) & -\nu(a+x) & (1-\nu)(b-y)/2 \\ \nu(b-y) & (b-y) & -(1-\nu)(a+x)/2 \\ (a-x) & \nu(a-x) & -(1-\nu)(b+y)/2 \\ -\nu(b+y) & -(b+y) & (1-\nu)(a-x)/2 \\ (a+x) & \nu(a+x) & (1-\nu)(b+y)/2 \\ \nu(b+y) & (b+y) & (1-\nu)(a+x)/2 \end{bmatrix} \quad \ldots(4.22)$$

Post multiplying this product by b we obtain the matrix (4.23). Integrating this between the limits $-a$ to a and $-b$ to b we get the stiffness matrix (4.24).

4.4.3 Criticism of the type of element derived in 4.4.2

Zienkiewicz (30) gives a simple example of a cantilever beam to which various elements are applied. His tabulated comparisons are reproduced here at Fig. 4.7

4.4.3.1 It can be easily seen that at a line of symmetry as shown in Fig. 4.8, an element ABCD will take up a shape AB'C'D as shown in Fig. 4.9. Angle DAB' is no longer a right angle, but it is axiomatic that angle DAB' must in reality remain a right angle. In complying with the requirements of a straight sided element, it is necessary to have the throat angle change shape. This introduces into

$$
\begin{array}{|c|c|c|c|c|c|c|c|}
\hline
(a-x)^2 + \frac{(1-\nu)}{2}(b-y)^2 & \frac{(1+\nu)}{2}\times(a-x)(b-y) & (a^2-x^2) - \frac{(1-\nu)}{2}(b-y)^2 & -\nu(a-x)(b-y) + \frac{(1-\nu)}{2}(a+x)(b-y) & -(a-x)^2 + \frac{(1-\nu)}{2}(b^2-y^2) & \nu(a-x)(b+y) - \frac{(1-\nu)}{2}(a-x)(b-y) & -(a^2-x^2) - \frac{(1-\nu)}{2}(b^2-y^2) & -\nu(a-x)(b+y) - \frac{(1-\nu)}{2}(a+x)(b-y) \\
\hline
 & (b-y)^2 + \frac{(1-\nu)}{2}(a-x)^2 & \nu(a+x)(b-y) - \frac{(1-\nu)}{2}(a-x)(b-y) & (b-y)^2 + \frac{(1-\nu)}{2}(a^2-x^2) & -\nu(a-x)(b-y) + \frac{(1-\nu)}{2}(a-x)(b+y) & (b-y)^2 - \frac{(1-\nu)}{2}(a-x)^2 & -\nu(a+x)(b-y) - \frac{(1-\nu)}{2}(a-x)(b+y) & -(b^2-y^2) - \frac{(1-\nu)}{2}(a^2-x^2) \\
\hline
 & & (a+x)^2 + \frac{(1-\nu)}{2}(b-y)^2 & -\frac{(1+\nu)}{2}\times(a+x)(b-y) & -(a^2-x^2) - \frac{(1-\nu)}{2}(b^2-y^2) & -(a^2-x^2) + \frac{(1-\nu)}{2}(a-x)(b-y) & -(a+x)^2 + \frac{(1-\nu)}{2}(b^2-y^2) & -\nu(a+x)(b+y) + \frac{(1-\nu)}{2}(a+x)(b-y) \\
\hline
 & & & (b-y)^2 + \frac{(1-\nu)}{2}(a+x)(b+y) & \nu(a-x)(b-y) + \frac{(1-\nu)}{2}(a+x)(b+y) & -(b^2-y^2) - \frac{(1-\nu)}{2}(a^2-x^2) & \nu(a+x)(b-y) - \frac{(1-\nu)}{2}(a+x)(b+y) & (b^2-y^2) - \frac{(1-\nu)}{2}(a+x)^2 \\
\hline
 & & & & (a-x)^2 + \frac{(1-\nu)}{2}(b+y)^2 & -\frac{(1+\nu)}{2}\times(a-x)(b+y) & (a^2-x^2) - \frac{(1-\nu)}{2}(b+y)^2 & \nu(a-x)(b+y) - \frac{(1-\nu)}{2}(b+y)(a+x) \\
\hline
 & & & & & (b+y)^2 + \frac{(1-\nu)}{2}(a-x)^2 & -\nu(a+x)(b+y) + \frac{(1-\nu)}{2}(a-x)(b+y) & -(b+y)^2 + \frac{(1-\nu)}{2}(a^2-x^2) \\
\hline
 & & & & & & (a+x)^2 + \frac{(1-\nu)}{2}(b+y)^2 & \frac{(1+\nu)}{2}\times(a+x)(b+y) \\
\hline
 & & & & & & & (b+y)^2 + \frac{(1-\nu)}{2}(a+x)^2 \\
\hline
\end{array}
$$

All terms to be multiplied by $Et/((4ab)^2(1-\nu^2))$

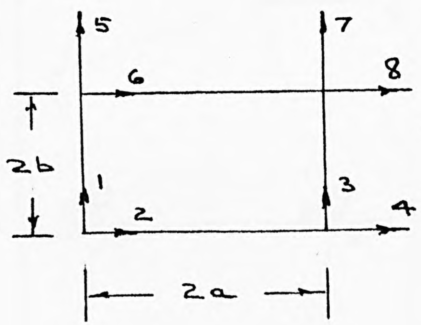
Terms symmetrical about leading diagonal.

Resulting matrix from the product $[b]^T[D][b]$ $\qquad \ldots(4.23)$

$\frac{a}{3b(1-\nu)}+\frac{b}{6a}$	$\frac{(1+\nu)}{8(1-\nu)}$	$\frac{a}{6b(1-\nu)}-\frac{b}{6a}$	$\frac{(1-3\nu)}{8(1-\nu)}$	$\frac{-a}{3b(1-\nu)}+\frac{b}{12a}$	$\frac{-(1-3\nu)}{8(1-\nu)}$	$\frac{-a}{6b(1-\nu)}-\frac{b}{12a}$	$\frac{-(1+\nu)}{8(1-\nu)}$
	$\frac{b}{3a(1-\nu)}+\frac{a}{6b}$	$\frac{-(1-3\nu)}{8(1-\nu)}$	$\frac{-b}{3a(1-\nu)}+\frac{a}{12b}$	$\frac{(1-3\nu)}{8(1-\nu)}$	$\frac{b}{6a(1-\nu)}-\frac{a}{6b}$	$\frac{-(1+\nu)}{8(1-\nu)}$	$\frac{-b}{6a(1-\nu)}-\frac{a}{12b}$
		$\frac{a}{3b(1-\nu)}+\frac{b}{6a}$	$\frac{-(1+\nu)}{8(1-\nu)}$	$\frac{-a}{6b(1-\nu)}-\frac{b}{12a}$	$\frac{(1+\nu)}{8(1-\nu)}$	$\frac{-a}{3b(1-\nu)}+\frac{b}{12a}$	$\frac{(1-3\nu)}{8(1-\nu)}$
			$\frac{b}{3a(1-\nu)}+\frac{a}{6b}$	$\frac{(1+\nu)}{8(1-\nu)}$	$\frac{-b}{6a(1-\nu)}-\frac{a}{12b}$	$\frac{-(1-3\nu)}{8(1-\nu)}$	$\frac{b}{6a(1-\nu)}-\frac{a}{6b}$
				$\frac{a}{3b(1-\nu)}+\frac{b}{6a}$	$\frac{-(1+\nu)}{8(1-\nu)}$	$\frac{a}{6b(1-\nu)}-\frac{b}{6a}$	$\frac{-(1-3\nu)}{8(1-\nu)}$
					$\frac{b}{3a(1-\nu)}+\frac{a}{6b}$	$\frac{(1-3\nu)}{8(1-\nu)}$	$\frac{-b}{3a(1-\nu)}+\frac{a}{12b}$
						$\frac{a}{3b(1-\nu)}+\frac{b}{6a}$	$\frac{(1+\nu)}{8(1-\nu)}$
							$\frac{b}{3a(1-\nu)}+\frac{a}{6b}$

All terms to be multiplied by $Et/(1+\nu)$

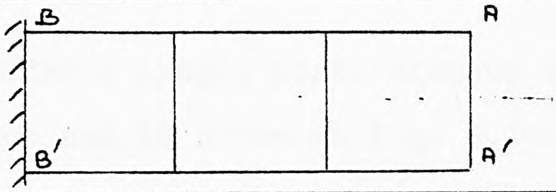
Terms symmetrical about leading diagonal.

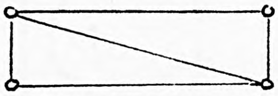
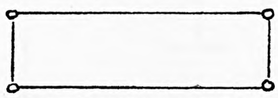
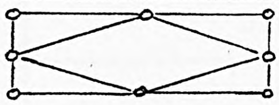
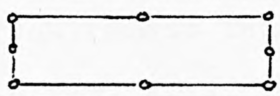
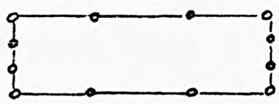


Order of degrees of freedom.

Plate stiffness matrix for a straight sided conforming element.

$$\ldots(4.24)$$

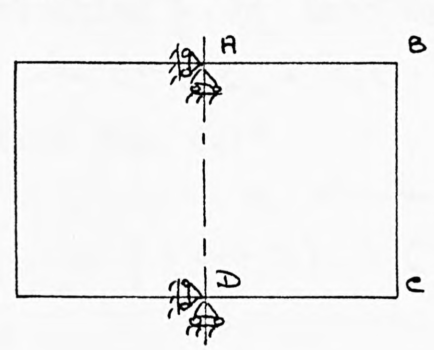


Type of Element	Vertical Load of A			
	Max. def. at AA'	Max. stress BB'	Max. def. at AA'	Max. stress BB'
	0.26	0.19	0.22	0.22
	0.65	0.56	0.67	0.67
	0.53	0.51	0.52	0.55
	0.99	0.99	1.00	1.00
	1.00	1.00	1.00	1.00
Exact	1.00	1.00	1.00	1.00

A cantilever in plane stress analysed by various elements. Accuracy improvement with higher order elements.

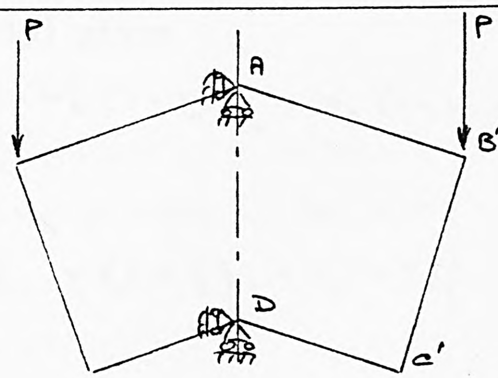
Fig. 4.7

the solution a non existant shear stress.



Symmetrical structure before loading.

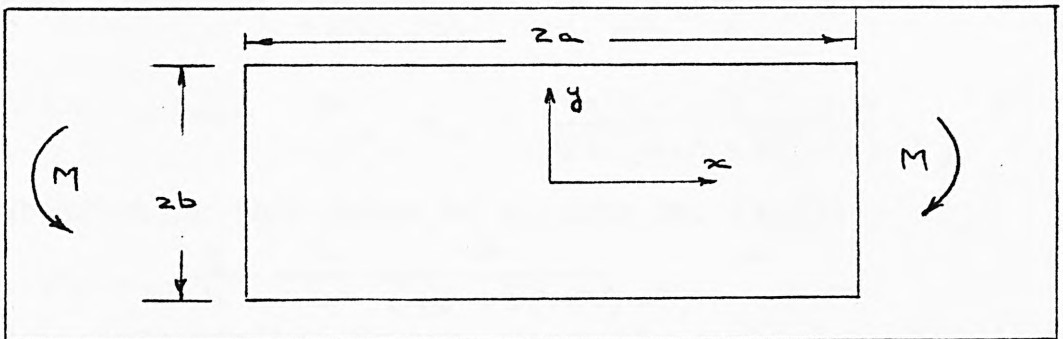
Fig. 4.8



Symmetrical structure after loading.

Fig. 4.9

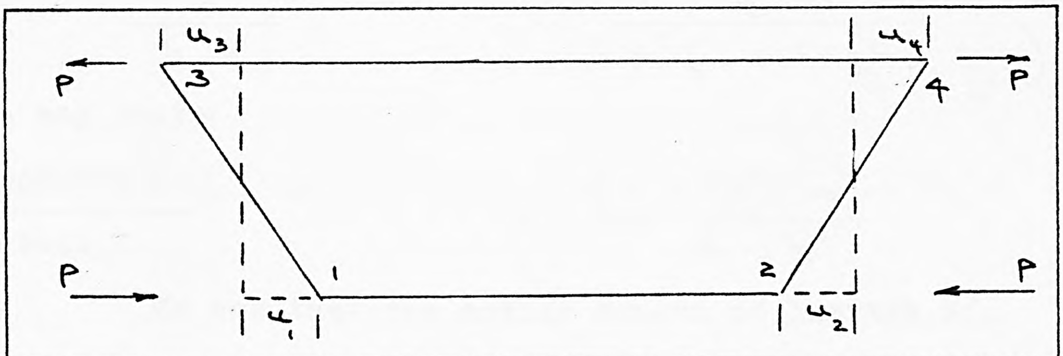
4.4.3.2 Consider a single plate element subjected to pure bending at each end as shown in Fig. 4.10. After loading the distorted shape which it takes up is dictated by the



Simple plate element before loading.

Fig. 4.10

requirement that all the sides remain straight and is shown in Fig. 4.11. The loading moment M is replaced by the loads P such that M = 2*b*P.



Simple plate element after loading.

Fig. 4.11

Substituting $\eta = y/b$ into Eq. (4.19) gives

$$e_x = -u_1(1-\eta)/4a + u_2(1-\eta)/4a - u_3(1+\eta)/4a + u_4(1+\eta)/4a$$

Now from Fig. 4.11

$$u_1 = u \qquad u_2 = -u \qquad u_3 = -u \qquad u_4 = u$$

$$\therefore e_x = \frac{u}{4a}\left\{-(1-\eta) - (1-\eta) + (1+\eta) + (1+\eta)\right\}$$

hence

$$e_x = \frac{u}{4a} \cdot 4 \cdot \eta = \frac{u}{a} \cdot \frac{y}{b}$$

Similarly Eq. (4.18) yields

$$e_y = 0$$

Substituting these strains into Eq. (4.8) yields

$$\sigma_x = \frac{E}{(1-\nu^2)} \cdot \frac{y}{ab} \, u_x \qquad \ldots(4.25)$$

From Eq.(4.24)

$$\text{any } P_x = \left(\frac{b}{3a(1-\nu)} + \frac{a}{6b} \right) u_x$$

$$\therefore \text{ for a load } P, \; u_x = \frac{12(1-\nu^2)\, ab}{Et(4b^2 + 2(1-\nu)a^2)} P.$$

Substituting this value of u_x into Eq. (4.25)

$$\sigma_x = \frac{y}{ab} \cdot \frac{12}{t} \cdot \frac{ab}{(4b^2 + 2(1-\nu)a^2)} P$$

Engineer's bending theory, where the applied bending moment is 2bP gives

$$\sigma_x = \frac{2bP\,y}{t\,8b^2/12}$$

Comparing these we find

	E.B.T.	Straight sided element
$\sigma_x =$	$\dfrac{3yP}{tb^2}$	$\dfrac{3yP}{tb^2\left(1 + \frac{(1-\nu)}{2}\frac{a^2}{b^2}\right)}$

We may write

$$\frac{\text{Apparent I}}{\text{Actual I}} = 1 + \frac{(1-\nu)\,a^2}{2\,b^2}$$

We see that the actual moment of inertia of each element's cross section is being effectively distorted by a factor of $\left\{ 1 + \frac{(1-\nu)}{2} \frac{a^2}{b^2} \right\}$

4.4.3.3 The four node plate element whose stiffness matrix has just been derived was chosen because it is the one most often given for this type. Its poor showing when dealing with simple bending made it necessary to find or create a more reliable element. The direct cause of its poor showing has just been shown to be due to individual elements appearing to have a greater moment of inertia and lower shear stiffness than is true. In deriving this type of plate stiffness matrix

a set of equations is formed relating displacements anywhere in or on the edges of the element to the displacements at the nodes. It follows that since any side has only two nodes and that as each node has only two degrees of freedom then there can only be a linear relationship between displacements along an edge and at that edge's nodes.
In the situation where two elements have two adjacent nodes common, then since there is only a linear relationship between position and displacement, the edge joining the two nodes must be a straight line and thus the elements are conforming. But since the displaced element can have only straight sides it is unable to map a smooth curve. Zienkiewicz (30) argues that non-conforming functions could be used to improve the performance of elements where conformity is easily obtained as well as those where it is not so easy. He deals with the question of how such a function could be generated.
A formal approach is possibly utilizing substitute shape functions. An approach on this line was made by Wilson et al (31) who introduced two additional displacement nodes. High order elements e.g. Zienkiewicz's isoparametric could be used. With both this and the Wilson element the penalty is an increase in core size to accommodate the stiffness matrix. In these cases though there is a possible advantage in that the number of elements might be reduced, but the number of degrees of freedom will be increased, thus perhaps giving little reduction in core size. The construction of a structure may be such that these gains vanish as more elements are required to represent the structure. Any increase of core space is

against the whole object of trying to reduce core space.

There now exist sufficient element types to cover almost every situation and there is no merit in formulating an element when there is one available though little used, which gives better results while using less core space than either the Zienkiewicz or Wilson elements.

Zienkiewicz gives physical intuition as another way of forming non-conforming functions. The derivation of the plate stiffness matrix which was finally chosen is now derived.

4.4.4 Derivation of second plate element.

4.4.4.1 In the derivation of this matrix intuition seems to have provided the initial concept. The following shows that the matrix can be evolved more logically by ammending part of the reasoning put forward by Przemieniecki (32) when he obtained another though less accurate rectangular plate element stiffness matrix. In this reasoning he showed that it was not necessary to assume a displacement distribution. If a stress distribution was chosen, a strain pattern could be obtained which if integrated would give a displacement distribution.

4.4.4.2 The assumed stress relationship he chose was linear so that

$$\sigma_x = a_8 + a_6 y$$

$$\sigma_y = a_7 + a_5 x$$

and $\tau_{xy} = a_4$

which leads to

$$e_x = \frac{1}{E}(a_8 + a_6 y - \nu a_7 - \nu a_5 x) \quad \ldots(4.26)$$

$$e_y = \frac{1}{E}(a_7 + a_5 x - \nu a_8 - \nu a_6 y) \quad \ldots(4.27)$$

$$e_{xy} = 2(1+\nu)a_4/E \quad \ldots(4.28)$$

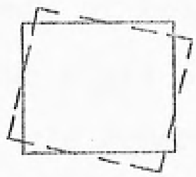
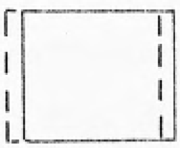
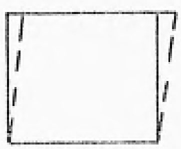
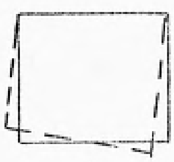
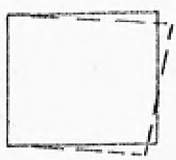
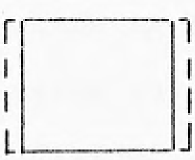
since $\partial u_x/\partial x = e_x$ and $\partial u_y/\partial y = e_y$

then Eq.(4.26) and Eq. (4.27) may be integrated to get

$$u_x = \frac{1}{E}\left(a_8 x + a_6 xy - \nu a_7 x - \frac{\nu a_5 x^2}{2}\right) + \text{Const } 3 \quad \ldots(4.29)$$

$$u_y = \frac{1}{E}\left(a_7 y + a_5 xy - \nu a_8 y - \frac{\nu a_6 y^2}{2}\right) + \text{Const } 2 \quad \ldots(4.30)$$

for this plate element there are two degrees of freedom at each of four nodes, therefore there must be eight coefficients to get a solution with eight displacements. At this stage there are five coefficients available, to obtain three more Przemieniecki arranges for Const 3 to be an arbitary function in y only. By substituting Eq. (4.29) and Eq. (4.30) into Eq. (4.28) values are obtained for the constants, which brings in three more coefficients a_1, a_2 and a_3. Przemieniecki writes that they represent rigid body rotation and the two rigid body translations respectively. This can be true of the translations since the extra terms introduced by the arbitary functions are independent of any terms with respect to which the displacement equation must be differentiated to obtain the strains e_x and e_y. However performing the operation $\partial u_x/\partial y + \partial u_y/\partial x$ to obtain shear strain will introduce terms which violate Eq. (4.28). Therefore rigid body rotation using this concept introduces shear strain into the element.

Physical interpretation of the terms in vector a		
a_1		Rigid body rotation.
a_2		Rigid body motion in the y direction.
a_3		Rigid body motion in the x direction.
a_4		Shear deformation.
a_5		Bending of the y axis.
a_6		Bending of the x axis.
a_7		Extension in y direction.
a_8		Extension in x direction.

Distortion modes of a rectangular element.

Fig. 4.12

4.4.4.3 Returning to Przemieniecki's solution but adding simple constants of integration at (4.29) and (4.30) will still leave one rigid body rotation constant missing. Consider a single general equation of the form of (4.29) and (4.30) with the missing constant a_1 arbitarily added. so that

$$u_i = \frac{1}{E}\left(N_{i1}a_1 + N_{i2}a_2 + N_{i3}a_3 + N_{i4}a_4 + N_{i5}a_5 + N_{i6}a_6 + N_{i7}a_7 + N_{i8}a_8\right)$$

where N is an elastic distortion function.
Using the a coefficient numbering based on Przemieniecki's work, shown in Fig. 4.12 and the freedom order of Fig. 4.5, and considering only nodal deflections the values of the matrix $[N]$ where $u = 1/E\ [N]\ a$ is given in Eq. (4.31)

$$\begin{bmatrix} a & 1 & 0 & 0 & a & 0 & -2a & 0 \\ -b & 0 & 1 & 0 & 0 & b & 0 & -2b \\ -a & 1 & 0 & 0 & -a & 0 & -2a & 0 \\ -b & 0 & 1 & 0 & 0 & -b & 0 & 2b \\ a & 1 & 0 & 0 & -a & 0 & 2a & 0 \\ b & 0 & 1 & 2b & 0 & -b & 0 & -2b \\ -a & 1 & 0 & 0 & a & 0 & 2a & 0 \\ b & 0 & 1 & 2b & 0 & b & 0 & 2b \end{bmatrix} \qquad \ldots(4.31)$$

It can be seen that columns 4 to 8 can be simplified by changing the associated variable and that further the E value outside may be made to vanish by considering it divided into all the eight variables.
So that we may now write Eq. (4.32).
Which is of the form $\quad u = [A]\ a \qquad \ldots(4.33)$

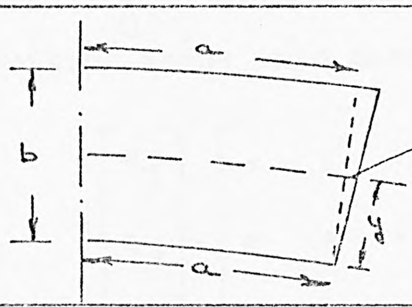
$$
\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \end{bmatrix}
=
\begin{bmatrix}
a & 1 & 0 & 0 & 1 & 0 & -1 & 0 \\
-b & 0 & 1 & 0 & 0 & 1 & 0 & -1 \\
-a & 1 & 0 & 0 & -1 & 0 & -1 & 0 \\
-b & 0 & 1 & 0 & 0 & -1 & 0 & 1 \\
a & 1 & 0 & 0 & -1 & 0 & 1 & 0 \\
b & 0 & 1 & 1 & 0 & -1 & 0 & -1 \\
-a & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\
b & 0 & 1 & 1 & 0 & 1 & 0 & 1
\end{bmatrix}
\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \end{bmatrix}
\quad \ldots(4.32)
$$

Where the terms of the vector are associated with a particular mode of distortion.

Term	Association
a_1	Rigid body rotation
a_2	Rigid body translation in the y direction
a_3	Rigid body translation in the x direction
a_4	Shear deformation
a_5	Rotations away from the y axis
a_6	Rotations away from the x axis
a_7	Translations in the y direction
a_8	Translations in the x direction

Derivation of the strain distribution.

Strain in the x direction due to bending

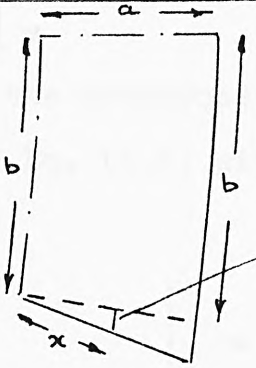


$$\text{increase} = \frac{a_6}{b} \cdot y$$

$$\therefore \text{strain} = \frac{a_6 \cdot y}{a \cdot b}$$

Fig. 4.13

Strain in the x direction due to Poisson's ratio effect.



increase $= \frac{a_5}{a} \cdot x$

leading to $-\nu \frac{a_5}{ab} \cdot x$ strain in the x direction.

Fig. 4.14

Strain in the x direction due to translation.

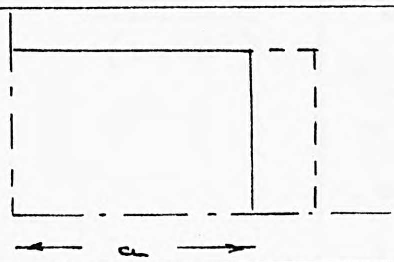


Fig. 4.15

Strain in the x direction due to translation which includes the poisson's ratio effect of extension in the y direction is a_8/a

Summing to get total strain in the x direction

$$e_x = a_6 y/ab - \nu a_5 x/ab + a_8/a$$

Similarly

$$e_y = a_5 x/ab - \nu a_6 y/ab + a_7/b$$

and

$$e_{xy} = a_4/2b$$

Putting these into matrix form

$$\begin{bmatrix} e_y \\ e_x \\ e_{xy} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & \frac{x}{ab} & \frac{-\nu y}{ab} & \frac{1}{b} & 0 \\ 0 & 0 & 0 & 0 & \frac{-\nu x}{ab} & \frac{y}{ab} & 0 & \frac{1}{a} \\ 0 & 0 & 0 & \frac{1}{2b} & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \end{bmatrix} \quad \ldots(4.34)$$

which is of the form

$$e = [B]\ a \qquad \ldots(4.35)$$

Post multiplying the transpose of this matrix $[B]$ by the matrix $[D]$ of Eq. (4.8) gives

$$\frac{E}{(1-\nu^2)}\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{(1-\nu)}{4b} \\ \frac{x(1-\nu^2)}{ab} & 0 & 0 \\ 0 & \frac{y(1-\nu^2)}{ab} & 0 \\ \frac{1}{b} & \frac{\nu}{b} & 0 \\ \frac{\nu}{a} & \frac{1}{a} & 0 \end{bmatrix} \qquad \ldots(4.36)$$

Matrix of the product $[B]^T\ [D]$

Post multiplying this matrix product by $[B]$ of Eq. (4.33) gives

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{(1-\nu)}{8b^2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{x^2(1-\nu^2)}{a^2b^2} & \frac{-\nu xy(1-\nu^2)}{a^2b^2} & \frac{x(1-\nu^2)}{ab^2} & 0 \\ 0 & 0 & 0 & 0 & \frac{-\nu xy(1-\nu^2)}{a^2b^2} & \frac{y^2(1-\nu^2)}{a^2b^2} & 0 & \frac{y(1-\nu^2)}{a^2b} \\ 0 & 0 & 0 & 0 & \frac{x(1-\nu^2)}{ab^2} & 0 & \frac{1}{b^2} & \frac{\nu}{ab} \\ 0 & 0 & 0 & 0 & 0 & \frac{y(1-\nu^2)}{a^2b} & \frac{\nu}{ab} & \frac{1}{a^2} \end{bmatrix}$$

Matrix of the product $[B]^T\ [D]\ [B]$ $\qquad \ldots(4.37)$

Integrating Eq. (4.37)

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{(1-\nu)}{8b^2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{x^2(1-\nu^2)}{a^2\,b^2} & \frac{-\nu xy(1-\nu^2)}{a^2\,b^2} & \frac{x(1-\nu^2)}{a\,b^2} & 0 \\ 0 & 0 & 0 & 0 & \frac{-\nu xy(1-\nu^2)}{a^2\,b^2} & \frac{y^2(1-\nu^2)}{a^2\,b^2} & 0 & \frac{y(1-\nu^2)}{a^2\,b} \\ 0 & 0 & 0 & 0 & \frac{x(1-\nu^2)}{a\,b^2} & 0 & \frac{1}{b^2} & \frac{\nu}{ab} \\ 0 & 0 & 0 & 0 & 0 & \frac{y(1-\nu^2)}{a^2\,b} & \frac{\nu}{ab} & \frac{1}{a^2} \end{bmatrix}$$

Matrix of the integration $[B]^T\,[D][B]\,dx.dy.dz$ ...(4.38)

Substituting the limits x = +a to −a

and y = +b to −b

Eq. (4.38) becomes

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{a}{2b(1+\nu)} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{4a}{3b} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{4b}{3a} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{4a}{b(1-\nu^2)} & \frac{4\nu}{(1-\nu^2)} \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{4\nu}{(1-\nu^2)} & \frac{4b}{a(1-\nu^2)} \end{bmatrix}$$

Matrix $[C]$ ...(4.39)

Substituting Eq. (4.33) into Eq. (4.35) gives

$$e = [B]\,[A^{-1}]\,u \qquad \text{...(4.40)}$$

Substituting Eq. (4.40) into Eq. (4.8) gives

$$\sigma = [D]\,[B]\,[A^{-1}]\;u$$

$$\therefore \quad \sigma^{T} = u^{T}\,[A^{-1}]^{T}\,[B]^{T}\,[D]^{T}$$

Substituting these values into Eq. (4.3)

$$P_v^{T}\,u_a = \iiint u_v^{T}\,[A^{-1}]^{T}\,[B^{T}]\,[D]^{T}\,[B]\,[A^{-1}]\;u_a \; dx.\,dy.\,dz \qquad \ldots(4.41)$$

Substituting Eq. (4.39) into Eq. (4.41) gives for a constant thickness t

$$P_v^{T}\,u_a = E\,t\,u_v^{T} \iint [A^{-1}]^{T}\,[C]\,[A^{-1}]\;u_a \; dx.\,dy \qquad \ldots(4.42)$$

Therefore as with the derivation of Eq. (4.9)

$$k = E\,t\,[A^{-1}]^{T}\,[C]\,[A^{-1}] \qquad \ldots(4.43)$$

Inverting Eq. (4.32) by hand gives

$$\begin{bmatrix}
1/4a & 0 & -1/4a & 0 & 1/4a & 0 & -1/4a & 0 \\
1/4 & 0 & 1/4 & 0 & 1/4 & 0 & 1/4 & 0 \\
b/4a & 1/2 & -b/4a & 1/2 & b/4a & 0 & -b/4a & 0 \\
-b/2a & -1/2 & b/2a & -1/2 & -b/2a & 1/2 & b/2a & 1/2 \\
1/4 & 0 & -1/4 & 0 & -1/4 & 0 & 1/4 & 0 \\
0 & 1/4 & 0 & -1/4 & 0 & -1/4 & 0 & 1/4 \\
-1/4 & 0 & -1/4 & 0 & 1/4 & 0 & 1/4 & 0 \\
0 & -1/4 & 0 & 1/4 & 0 & -1/4 & 0 & 1/4
\end{bmatrix}$$

$$\text{Matrix } \left[A^{-1}\right] \qquad \ldots(4.44)$$

Transposing Eq. (4.44) gives

$$\begin{bmatrix}
\frac{1}{4a} & \frac{1}{4} & \frac{b}{4a} & -\frac{b}{2a} & \frac{1}{4} & 0 & -\frac{1}{4} & 0 \\
0 & 0 & \frac{1}{2} & -\frac{1}{2} & 0 & \frac{1}{4} & 0 & -\frac{1}{4} \\
-\frac{1}{4a} & \frac{1}{4} & -\frac{b}{4a} & \frac{b}{2a} & -\frac{1}{4} & 0 & -\frac{1}{4} & 0 \\
0 & 0 & \frac{1}{2} & -\frac{1}{2} & 0 & -\frac{1}{4} & 0 & \frac{1}{4} \\
\frac{1}{4a} & \frac{1}{4} & \frac{b}{4a} & -\frac{b}{2a} & -\frac{1}{4} & 0 & \frac{1}{4} & 0 \\
0 & 0 & 0 & \frac{1}{2} & 0 & -\frac{1}{4} & 0 & -\frac{1}{4} \\
-\frac{1}{4a} & \frac{1}{4} & -\frac{b}{4a} & \frac{b}{2a} & \frac{1}{4} & 0 & \frac{1}{4} & 0 \\
0 & 0 & 0 & \frac{1}{2} & 0 & \frac{1}{4} & 0 & \frac{1}{4}
\end{bmatrix}$$

Matrix $\left[A^{-1}\right]^T$ ...(4.45)

Post multiplying Eq. (4.45) by Eq. (4.39)

$$\begin{bmatrix}
0 & 0 & 0 & \frac{-1}{4(1+\nu)} & \frac{a}{3b} & 0 & \frac{-a}{b(1-\nu^2)} & 0 \\
0 & 0 & 0 & \frac{-a}{4b(1+\nu)} & 0 & \frac{b}{3a} & 0 & \frac{-b}{a(1-\nu^2)} \\
0 & 0 & 0 & \frac{1}{4(1+\nu)} & -\frac{a}{3b} & 0 & \frac{-a}{b(1-\nu^2)} & 0 \\
0 & 0 & 0 & \frac{-a}{4b(1+\nu)} & 0 & \frac{-b}{3a} & 0 & \frac{b}{a(1-\nu^2)} \\
0 & 0 & 0 & \frac{-1}{4(1+\nu)} & \frac{-a}{3b} & 0 & \frac{a}{b(1-\nu^2)} & 0 \\
0 & 0 & 0 & \frac{a}{4b(1+\nu)} & 0 & \frac{-b}{3a} & 0 & \frac{-b}{a(1-\nu^2)} \\
0 & 0 & 0 & \frac{1}{4(1+\nu)} & \frac{a}{3b} & 0 & \frac{a}{b(1-\nu^2)} & 0 \\
0 & 0 & 0 & \frac{a}{4b(1+\nu)} & 0 & \frac{b}{3a} & 0 & \frac{b}{a(1-\nu^2)}
\end{bmatrix}$$

Matrix $\left[A^{-1}\right]^T \left[C\right]$ ...(4.46)

$\frac{a}{12b}+\frac{a}{4b(1-\nu^2)}+\frac{b}{8a(1+\nu)}$	$\frac{\nu}{4(1-\nu^2)}+\frac{1}{8(1+\nu)}$	$\frac{-a}{12b}+\frac{a}{4b(1-\nu^2)}-\frac{b}{8a(1+\nu)}$	$\frac{-\nu}{4(1-\nu^2)}+\frac{1}{8(1+\nu)}$	$\frac{-a}{12b}-\frac{a}{4b(1-\nu^2)}+\frac{b}{8a(1+\nu)}$	$\frac{\nu}{4(1-\nu^2)}-\frac{1}{8(1+\nu)}$	$\frac{a}{12b}-\frac{a}{4b(1-\nu^2)}-\frac{b}{8a(1+\nu)}$	$\frac{-\nu}{4(1-\nu^2)}-\frac{1}{8(1+\nu)}$
	$\frac{b}{12a}+\frac{b}{4a(1-\nu^2)}+\frac{a}{8b(1+\nu)}$	$\frac{\nu}{4(1-\nu^2)}-\frac{1}{8(1+\nu)}$	$-\frac{b}{12a}-\frac{b}{4a(1-\nu^2)}+\frac{a}{8b(1+\nu)}$	$-\frac{\nu}{4(1-\nu^2)}+\frac{1}{8(1+\nu)}$	$-\frac{b}{12a}+\frac{b}{4a(1-\nu^2)}-\frac{a}{8b(1+\nu)}$	$\frac{-\nu}{4(1-\nu^2)}-\frac{1}{8(1+\nu)}$	$\frac{b}{12a}-\frac{b}{4a(1-\nu^2)}-\frac{a}{8b(1+\nu)}$
		$\frac{a}{12b}+\frac{a}{4b(1-\nu^2)}+\frac{b}{8a(1+\nu)}$	$-\frac{\nu}{4(1-\nu^2)}-\frac{1}{8(1+\nu)}$	$\frac{a}{12b}-\frac{a}{4b(1-\nu^2)}-\frac{b}{8a(1+\nu)}$	$\frac{\nu}{4(1-\nu^2)}+\frac{1}{8(1+\nu)}$	$-\frac{a}{12b}-\frac{a}{4b(1-\nu^2)}+\frac{b}{8a(1+\nu)}$	$-\frac{\nu}{4(1-\nu^2)}+\frac{1}{8(1+\nu)}$
			$\frac{b}{12a}+\frac{b}{4a(1-\nu^2)}+\frac{a}{8b(1+\nu)}$	$\frac{\nu}{4(1-\nu^2)}+\frac{1}{8(1+\nu)}$	$\frac{b}{12a}-\frac{b}{4a(1-\nu^2)}-\frac{a}{8b(1+\nu)}$	$\frac{\nu}{4(1-\nu^2)}-\frac{1}{8(1+\nu)}$	$-\frac{b}{12a}+\frac{b}{4a(1-\nu^2)}-\frac{a}{8b(1+\nu)}$
				$\frac{a}{12b}+\frac{a}{4b(1-\nu^2)}+\frac{b}{8a(1+\nu)}$	$-\frac{\nu}{4(1-\nu^2)}-\frac{1}{8(1+\nu)}$	$-\frac{a}{12b}+\frac{a}{4b(1-\nu^2)}-\frac{b}{8a(1+\nu)}$	$\frac{\nu}{4(1-\nu^2)}-\frac{1}{8(1+\nu)}$
					$\frac{b}{12a}+\frac{b}{4a(1-\nu^2)}+\frac{a}{8b(1+\nu)}$	$-\frac{\nu}{4(1-\nu^2)}+\frac{1}{8(1+\nu)}$	$-\frac{b}{12a}-\frac{b}{4a(1-\nu^2)}+\frac{a}{8b(1+\nu)}$
						$\frac{a}{12b}+\frac{a}{4b(1-\nu^2)}+\frac{b}{8a(1+\nu)}$	$\frac{\nu}{4(1-\nu^2)}+\frac{1}{8(1+\nu)}$
							$\frac{b}{12a}+\frac{b}{4a(1-\nu^2)}+\frac{a}{8b(1+\nu)}$

All terms to be multiplied by Et

Terms symmetrical about leading diagonal.

Order of degrees of freedom as (4.24)

Plate stiffness matrix derived using distortion modes.

...(4.47)

$U.(H.F+V)$	VP_i	$U.(H.FF-V)$	$-VM_i$	$U.(-H.F+V)$	VM_i	$U.(-H.FF-V)$	$-VP_i$
	$U.(V.F+H)$	VM_i	$U.(-V.F+H)$	$-VM_i$	$U.(V.FF-H)$	$-VP_i$	$U.(-V.FF-H)$
		$U.(H.F+V)$	VP_i	$U.(-H.FF-V)$	VP_i	$U.(-H.F+V)$	VM_i
			$U.(V.F+H)$	VP_i	$U.(-V.FF-H)$	VM_i	$U.(V.FF-H)$
				$U.(H.F+V)$	$-VP_i$	$U.(H.FF-V)$	VM_i
					$U.(V.F+H)$	$-VM_i$	$U.(-V.F+H)$
						$U.(H.F+V)$	VP_i
							$U.(V.F+H)$

where

t_i is the thickness of the i^{th} element.

X_i is $X.t_i$ and Y_i is $Y.t_i$

U is $E/8(1+\nu)$

H is X_i/Y and V is Y_i/X

F is $(1/12 + 1/4(1-\nu^2)).8(1+\nu)$

FF is $(1/12 - 1/4(1-\nu^2)).8(1+\nu)$

VP_i is $(\nu/4(1-\nu^2) + 1/8(1+\nu)).E.t_i$

VM_i is $(\nu/4(1-\nu^2) - 1/8(1+\nu)).E.t_i$

The stiffness matrix for rectangular elements obtained using distortion modes and given in (4.47), modified to take account of assembly problems.

...(4.48)

Fig. 4.16

Post multiplying Eq. (4.46) by Eq. (4.44) gives the plate element stiffness matrix used in this package as shown in Eq. (4.47). This form is not the best arrangement for the assembly technique used and Eq. (4.47) is re-expressed as shown in Eq. (4.48).

Using this element the results shown in Fig. 4.7 for a rectangular plate element with four nodes becomes much nearer the exact solution as shown in example 1 of appendix C.

4.5 Derivation of a triangular plate element.

4.5.1 Minimising the number of triangular elements.

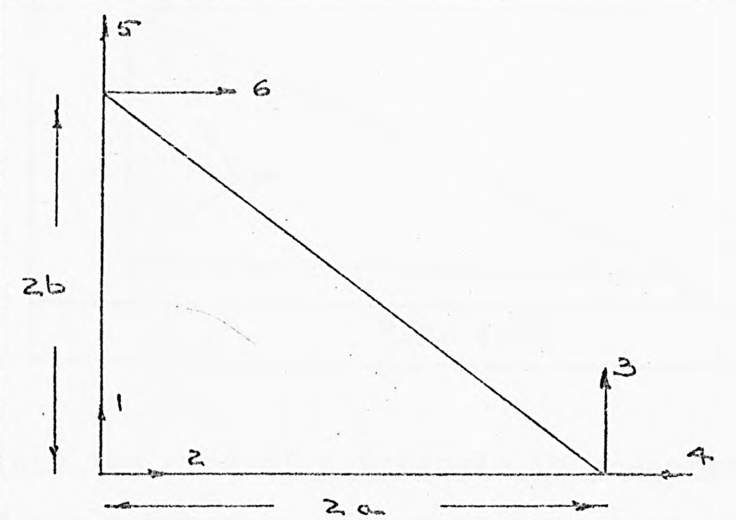
4.5.1.1 It can be seen from Fig. 4.7 that simple triangular elements are best avoided if possible. For this reason triangular panels are idealised to maximise the use of rectangular elements. Olson and Bearden (33) suggest that this type of element is still competitive. In this work only right angled triangular panels are acceptable, if a triangular panel cannot be avoided. In such cases triangular panels are automatically dealt with as a series of rectangular elements with just the last element on each row being handled as a triangular element.

4.5.2 Corollary with plate elements.

4.5.2.1 In the derivation of the first type of rectangular plate element a relationship was established between any random displacement in the element and the element's node displacements using a set of linear shape functions.

This enabled the matrix $[B]$ to relate strains to nodal displacements directly without reference to a second matrix $[A]$ formed from the coefficients of the equations assumed to relate displacements within the element to nodal displacements from which strains could then be found. This enabled the Eq. (4.9) to contain only two individual matrices. To use the matrix $[A]$ we see that by expanding Eq. (4.43), Eq. (4.9) becomes

$$k = \frac{Et}{(1-\nu^2)} \iint A^{-1\,T} \; B \; D \; B \; A^{-1} \, dx.dy \qquad \ldots(4.49)$$



Order of the numbering of degrees of freedom for a triangular element to be consistent with the rectangular element developed in 4.3

Fig. 4.17

There are obvious disadvantages here in that there is an extra individual matrix to be formed which has to be inverted and a doubling of the number of matrix multiplications. The following derivation of a triangular plate matrix avoids these disadvantages by generating shape functions as in 4.3.1

4.5.2.2 In the case of a triangular element, the formation of the shape functions as used in 4.3.1 are unable to accommodate the hypoteneuse. This is because those shape functions are designed to operate linearly along edges parallel to the local axes. In triangular elements resort can be made to a system of area co-ordinates.

4.5.2.3 Consider the right angled triangle 123. If point P moves parallel to the side 12 the area of triangle 1P3 will change in direct proportion to the distance moved,

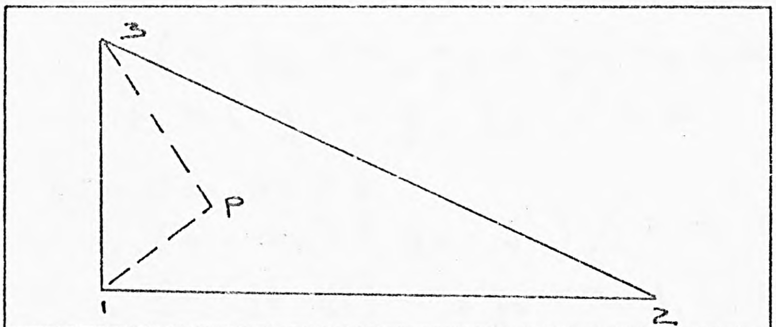


Fig. 4.18

since the area of a triangle is proportional to its height. If the point P moves parallel to side 13 then there is no change in the area of triangle 1p3. It then follows that if P moves in any direction there will exist a linear relationship between the areas 1P3 and 123.

Let L_1 = area 2P3 / area 123

L_2 = area 3P1 / area 123

and L_3 = area 1P2 / area 123

The L ratios for a right angled triangle may be found as follows

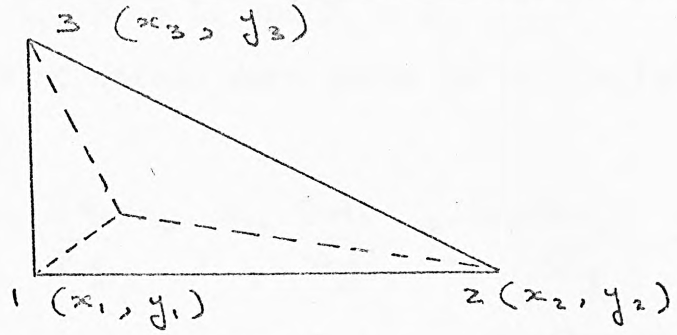


Fig. 4.19

$$\text{Area } 2P3 = (x_2 - x_3).(y_3 + y_2)/2$$
$$- (x - x_3).(y + y_3)/2$$
$$- (x_2 - x).(y + y_2)/2$$

Let the area of triangle 123 be A,

then L_1 = area 2P3 / A

$$= (x_2 y_3 - x_3 y_2 + y(x_3 - x_2) + x(y_2 - y_3))/2A \quad \ldots(4.50)$$

L_2 = area 3P1 / A

$$= (x - x_3)(y_3 - y_1)/2A$$
$$= (x(y_3 - y_1) - x_3 y_3 + x_3 y_1)/2A \quad \ldots(4.51)$$

L_3 = area 1P2 / A

$$= (y - y_1)(x_2 - x_1)/2A$$
$$= (y(x_2 - x_1) - y_1 x_2 + y_1 x_1)/2A \quad \ldots(4.52)$$

The displacement of a general point within the right angled triangle shown in Fig. 4.17 is given in terms of its six nodal degrees of freedom as

$$u_x = N_1 u_2 + N_2 u_4 + N_3 u_6$$
$$u_y = N_1 u_1 + N_2 u_3 + N_3 u_5$$

where N_1, N_2 and N_3 are shape functions. For linearly varying displacement it can be seen that these take on the values of the area co-ordinates thus

$$N_1 = L_1, \; N_2 = L_2 \text{ and } N_3 = L_3$$

The components of strain were given in Eq. (4.12), Eq. (4.13) and Eq. (4.14)

$$\therefore \; e_y = u_1 \frac{\partial N_1}{\partial y} + u_3 \frac{\partial N_2}{\partial y} + u_5 \frac{\partial N_3}{\partial y} \qquad \ldots(4.53)$$

$$e_x = u_2 \frac{\partial N_1}{\partial x} + u_4 \frac{\partial N_2}{\partial x} + u_6 \frac{\partial N_3}{\partial x} \qquad \ldots(4.54)$$

$$e_{xy} = u_1 \frac{\partial N_1}{\partial x} + u_2 \frac{\partial N_1}{\partial y} + u_3 \frac{\partial N_2}{\partial x}$$
$$+ u_4 \frac{\partial N_2}{\partial y} + u_5 \frac{\partial N_3}{\partial x} + u_6 \frac{\partial N_3}{\partial y} \qquad \ldots(4.55)$$

Differentiating Eq. (4.50), Eq. (4.51) and Eq. (4.52) and substituting into Eq. (4.53) , Eq. (4.54) and Eq. (4.55) we obtain

$$e_y = (u_1 x_{32} + u_5 x_{21}) / 2A \qquad \ldots(4.56)$$

$$e_x = (-u_2 y_{32} + u_4 y_{31}) / 2A \qquad \ldots(4.57)$$

$$e_{xy} = (-u_1 y_{32} + u_3 y_{31}$$
$$+ u_2 x_{32} + u_6 x_{21}) / 2A \qquad \ldots(4.58)$$

where $x_{ij} = x_i - x_j$

and $y_{ij} = y_i - y_j$

The strain equations may be grouped together as

$$\begin{bmatrix} e_y \\ e_x \\ e_{xy} \end{bmatrix} = \frac{1}{2A} \begin{bmatrix} x_{32} & 0 & 0 & 0 & x_{21} & 0 \\ 0 & -y_{32} & 0 & y_{31} & 0 & 0 \\ -y_{32} & x_{32} & y_{31} & 0 & 0 & x_{21} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{bmatrix} \qquad \ldots(4.59)$$

which is of the form $e = [B] u$

Post multiplying $[B]^T$ by $[D]$ as defined in Eq. (4.6) and Eq. (4.5) we get

$$\frac{E}{2A(1-\nu^2)}\begin{bmatrix} x_{32} & \nu x_{32} & -y_{32}\frac{(1-\nu)}{2} \\ -\nu y_{32} & -y_{32} & x_{32}\frac{(1-\nu)}{2} \\ 0 & 0 & y_{31}\frac{(1-\nu)}{2} \\ \nu y_{31} & y_{31} & 0 \\ x_{21} & \nu x_{21} & 0 \\ 0 & 0 & x_{21}\frac{(1-\nu)}{2} \end{bmatrix} \qquad \ldots(4.60)$$

Post multiplying Eq. (4.60) by B gives Eq. (4.61). The stiffness matrix of the element is defined by Eq. (4.9) as

$$k = t \iint [B]^T [D] [B] \; dx.\,dy$$

where

$[B]^T [D] [B]$ is given by Eq. (4.61).

The integration is taken over the whole area of the triangle and as this matrix product is independent of any terms in x or y

$$k = \frac{A\,t}{4A^2(1-\nu^2)} [B]^T [D] [B]$$

where A is the area of the triangle.

4.5.3 Splitting the stiffness matrix.

4.5.3.1 For convenience of presentation the stiffness matrix can be separated into two parts, so that $k = k_n + k_s$ where k_n represents stiffness due to inplane stresses and

k_s represents stiffness due to shearing stresses.

$$\frac{E}{4A^2(1-\nu^2)}\left[\begin{array}{cccccc}
x_{32}^2+y_{32}^2\frac{(1-\nu)}{2} & -y_{32}x_{32}\nu-x_{32}y_{32}\frac{(1-\nu)}{2} & 0-y_{31}y_{32}\frac{(1-\nu)}{2} & y_{31}x_{32}\nu+0 & x_{21}x_{32}+0 & 0-x_{21}y_{32}\frac{(1-\nu)}{2} \\
-x_{32}y_{32}\nu-y_{32}x_{32}\frac{(1-\nu)}{2} & y_{32}^2+x_{32}^2\frac{(1-\nu)}{2} & 0+y_{31}x_{32}\frac{(1-\nu)}{2} & -y_{31}x_{32}+0 & -x_{21}y_{32}\nu+0 & 0+x_{21}x_{32}\frac{(1-\nu)}{2} \\
0-y_{32}y_{31}\frac{(1-\nu)}{2} & 0+x_{32}y_{31}\frac{(1-\nu)}{2} & 0+y_{31}^2\frac{(1-\nu)}{2} & 0+0 & 0+0 & 0+x_{21}y_{31}\frac{(1-\nu)}{2} \\
x_{32}y_{31}\nu+0 & -y_{32}y_{31}+0 & 0+0 & y_{31}^2+0 & x_{21}y_{31}\nu+0 & 0+0 \\
x_{32}x_{21}+0 & -y_{32}x_{21}\nu+0 & 0+0 & 0+0 & x_{21}^2+0 & 0+0 \\
0-y_{32}x_{21}\frac{(1-\nu)}{2} & 0+x_{32}x_{21}\frac{(1-\nu)}{2} & 0+y_{31}x_{21}\frac{(1-\nu)}{2} & y_{31}x_{21}\nu+0 & 0+0 & 0+x_{21}^2\frac{(1-\nu)}{2}
\end{array}\right] \quad \ldots(4.61)$$

4.5.3.2 The matrix product $[B]^T [D] [B]$

given in Eq. (4.61), k_n given in Eq. (4.62) and k_s given in Eq. (4.63) are not the values usually seen for a triangular element's stiffness matrix with inplane loading because this case is not a general case but applies only to right angled triangles.

$$k_n = \frac{Et}{2ab(1-\nu^2)} \begin{bmatrix} a^2 & \nu ab & 0 & -\nu ab & -a^2 & 0 \\ \nu ab & b^2 & 0 & -b^2 & -\nu ab & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -\nu ab & -b^2 & 0 & b^2 & \nu ab & 0 \\ -a^2 & -\nu ab & 0 & \nu ab & a^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \ldots(4.62)$$

$$k_s = \frac{Et}{4ab(1+\nu)} \begin{bmatrix} b^2 & ab & -b^2 & 0 & 0 & -ab \\ ab & a^2 & -ab & 0 & 0 & -a^2 \\ -b^2 & -ab & b^2 & 0 & 0 & ab \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -ab & -a^2 & ab & 0 & 0 & a^2 \end{bmatrix} \quad \ldots(4.63)$$

The basic plate element which was used for this package has been derived. Results obtained using this element have been compared with those obtained using well known elements to show that within the limits required its performance is as good as or better than more complex elements.

4.6 Beam element

4.6.1 There remains now the derivation of the stiffness matrix of the stiffeners to the plates. The matrix used is perfectly satisfactory for the problems it is required to deal with here and is a modification of the one shown in many books. The plate element matrix uses edge dimensions which are half the length of a side. To be consistent the beam element has been made in terms of half its length.

4.6.2 Consider the direct tensile and compressive loads on a beam, whose local axes and loading are shown in Fig. 4.20.

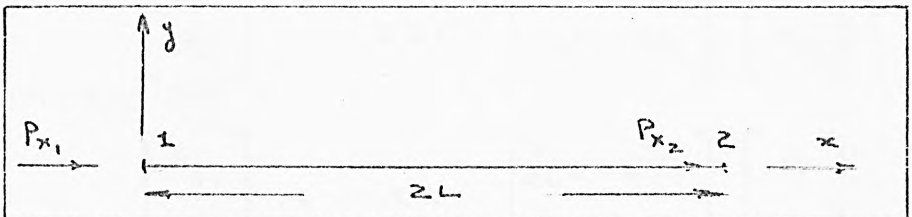


Fig. 4.20

If node 1 is displaced u_1 and node 2 is displaced u_2 along the local x axis. Then within the limits of elasticity we may write the relationship between strees and strain as

$$E = \frac{P_{x_1} / A}{(u_1 - u_2)/2L} \qquad \ldots(4.64)$$

hence $P_{x_1} = AE\,(u_1 - u_2)/2L$

$$\ldots(4.65)$$

similarly

$P_{x_2} = AE\,(-u_1 + u_2)/2L$

$$\ldots(4.66)$$

$$\begin{bmatrix} P_{x_1} \\ P_{x_2} \end{bmatrix} = \frac{AE}{2L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \qquad \ldots(4.67)$$

4.6.3 Now consider shearing forces and moments as shown in Fig. 4.21 causing displacement in the local y direction of v_i and rotation θ_i at node i.

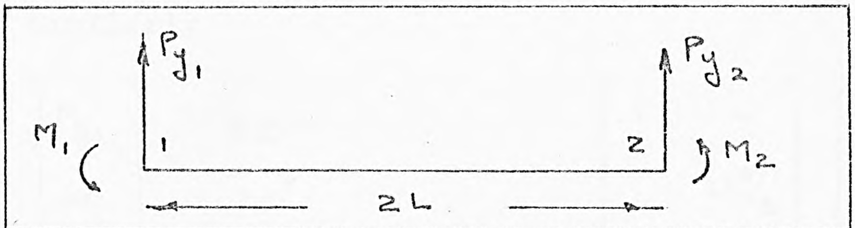


Fig. 4.21

Using the Moment-Area theorems to derive deflections and slopes we have

$$\begin{bmatrix} v_1 \\ \theta_1 \end{bmatrix} = \frac{1}{3IE} \begin{bmatrix} 8L^3 & -6L^2 \\ -6L^2 & 6L \end{bmatrix} \begin{bmatrix} P_{y_1} \\ M_1 \end{bmatrix} \qquad \ldots(4.68)$$

Solving for

$$\begin{bmatrix} P_{y_1} \\ M_1 \end{bmatrix} = IE \begin{bmatrix} \frac{3}{2L^3} & \frac{3}{2L^2} \\ \frac{3}{2L^2} & \frac{2}{L} \end{bmatrix} \begin{bmatrix} v_1 \\ \theta_1 \end{bmatrix} \qquad \ldots(4.69)$$

A similar relationship for the forces and displacements at node 2 may be obtained to give

$$\begin{bmatrix} P_{y_2} \\ M_2 \end{bmatrix} = IE \begin{bmatrix} \frac{3}{2L^3} & \frac{3}{2L^2} \\ \frac{3}{2L^2} & \frac{2}{L} \end{bmatrix} \begin{bmatrix} v_2 \\ \theta_2 \end{bmatrix} \qquad \ldots(4.70)$$

For equilibrium

$$P_{y_1} + P_{y_2} = 0$$

$$2P_{y_1}.L - M_1 - M_2 = 0$$

which may be written as

$$\begin{bmatrix} P_{y_2} \\ M_2 \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 2L & -1 \end{bmatrix} \begin{bmatrix} P_{y_1} \\ M_1 \end{bmatrix} \qquad \ldots(4.71)$$

Substituting Eq. (4.69) into Eq. (4.71)

$$\begin{bmatrix} P_{y_2} \\ M_2 \end{bmatrix} = \frac{3IE}{2L^3} \begin{bmatrix} -1 & 0 \\ 2L & -1 \end{bmatrix} \begin{bmatrix} 1 & L \\ L & 4L^2/3 \end{bmatrix} \begin{bmatrix} v_1 \\ \theta_1 \end{bmatrix}$$

$$= \frac{3IE}{2L^3} \begin{bmatrix} -1 & -L \\ L & 2L^2/3 \end{bmatrix} \begin{bmatrix} v_1 \\ \theta_1 \end{bmatrix} \quad \ldots(4.72)$$

Similarly

$$\begin{bmatrix} P_{y_1} \\ M_1 \end{bmatrix} = \frac{3IE}{2L^3} \begin{bmatrix} -1 & L \\ -L & 2L^2/3 \end{bmatrix} \begin{bmatrix} v_2 \\ \theta_2 \end{bmatrix} \quad \ldots(4.73)$$

4.6.4 Assembling these equations into an overall matrix.

$$\begin{bmatrix} P_{x_1} \\ P_{y_1} \\ M_1 \\ P_{x_2} \\ P_{y_2} \\ M_2 \end{bmatrix} = \frac{E}{2L} \begin{bmatrix} A & 0 & 0 & -A & 0 & 0 \\ 0 & \frac{3I}{L^2} & \frac{3I}{L} & 0 & -\frac{3I}{L^2} & \frac{3I}{L} \\ 0 & \frac{3I}{L} & 4I & 0 & -\frac{3I}{L} & 2I \\ -A & 0 & 0 & A & 0 & 0 \\ 0 & -\frac{3I}{L^2} & -\frac{3I}{L} & 0 & \frac{3I}{L^2} & -\frac{3I}{L} \\ 0 & \frac{3I}{L} & 2I & 0 & -\frac{3I}{L} & 4I \end{bmatrix} \begin{bmatrix} u_1 \\ v_1 \\ \theta_1 \\ u_2 \\ v_2 \\ \theta_2 \end{bmatrix}$$

$$\ldots(4.74)$$

Where A is the cross sectional area of the beam.

E is Young's modulus.

L is the half length of the beam element.

I is the beam's moment of inertia for bending about its centroid.

4.6.5 To simplify the input arrangements the only loads that will be allowed in the structure are those which fall on the nodes of a plate element.

The beam stiffness matrix just developed is for a load system applied at the beam's centroid. A transformation matrix is required to evaluate the effect of the structural loads on the plate node as they appear to the beam's node.

A load P_{p_y} or M_{p_z} when applied at the plate's node will appear to the stiffener to be acting through its node and have exactly the same effect and require no transformation. However a load at the plate's node P_{p_x} will appear to the stiffener's node as a load P_{s_x} equal to P_{p_x} and a moment M_{s_z} equal to $-P_{p_x} \cdot Ec$, where Ec is the eccentricity between the two nodes. We can therefore write

$$\begin{bmatrix} P_{p_x} \\ P_{p_y} \\ M_{p_z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -Ec & 0 & 1 \end{bmatrix} \begin{bmatrix} P_{s_x} \\ P_{s_y} \\ M_{s_z} \end{bmatrix} \qquad \ldots(4.75)$$

Which is of the form $P_p = a\, P_s$

where a is the transformation matrix. For the whole beam a becomes the following 6 x 6 matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ -Ec & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -Ec & 0 & 1 \end{bmatrix} \qquad \ldots(4.76)$$

Similarly for displacements a relationship between plate and stiffener nodes can be found.

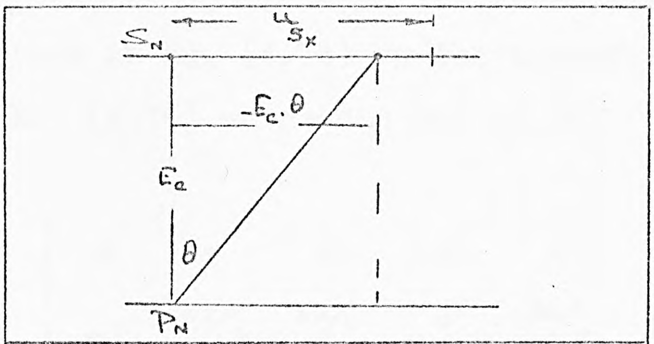


Fig. 4.22

From Fig. 4.22 we can see that if there is a dispalcement u_{s_x} of the stiffener node S_n there will be an equal displacement of P_n of u_{p_x} but the rotation of the vertical line between the node will cause u_{p_x} to be reduced by an amount Ec.θ. We can therefore write

$$\begin{bmatrix} u_{s_x} \\ u_{s_y} \\ \theta_{s_z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -E_c \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_{p_x} \\ u_{p_y} \\ \theta_{p_z} \end{bmatrix} \qquad \ldots(4.77)$$

Comparing this with Eq.(4.75) we see that this is of the form $u_s = [a^T]\, u_p$. For the whole beam this becomes a 6 x 6 matrix transpose of Eq. (4.76).

We have $\quad P_s = [K_s]\, U_s$

and $\quad U_s = [a^T]\, U_p$

$\therefore \quad P_s = [K_s][a^T]\, U_p$

$[a]\, P_s = [a][K_s][a^T]\, U_p$

but $\quad P_p = [a]\, P_s$

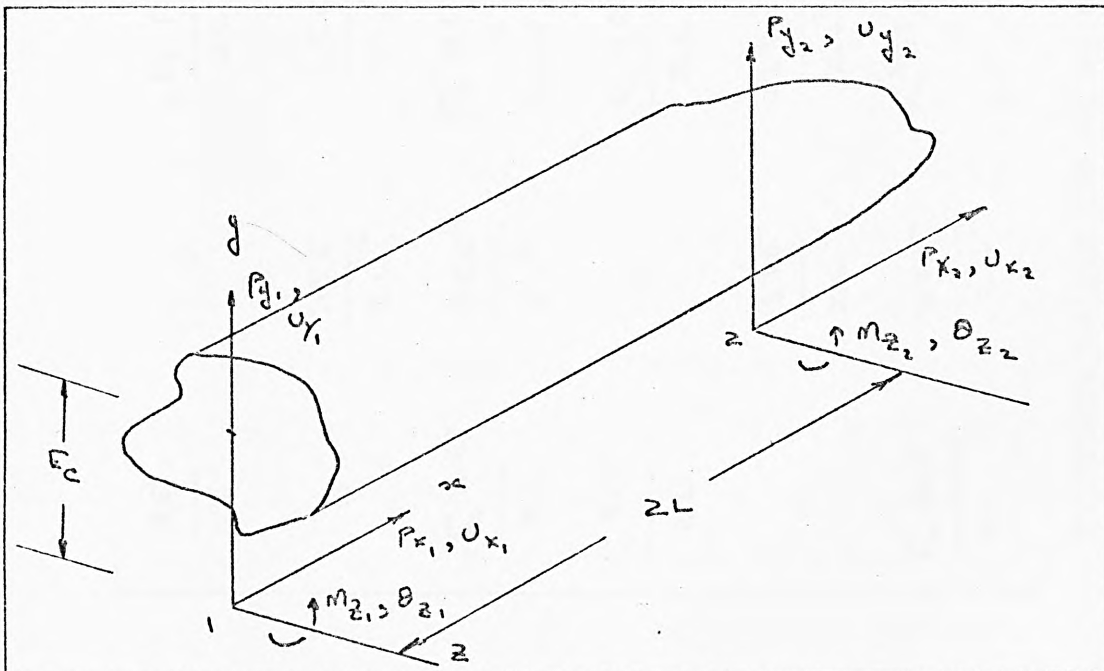
$\therefore \quad P_p = ([a][K_s][a^T])\, U_p$

Hence the transformation to be applied to a beam's stiffness matrix to obtain its revised stiffness matrix allowing for beam loading through the node of its associated plate is $[a][K_s][a^T]$

Premultiplying the beam's stiffness matrix as defined in Eq. (4.74) by the transformation matrix given in Eq. (4.76) we obtain Eq. (4.78)

$$\frac{E}{2L}\begin{bmatrix} A & 0 & 0 & -A & 0 & 0 \\ 0 & 3I/L^2 & 3I/L & 0 & -3I/L^2 & 3I/L \\ 0 & 3I/L & 4I & A.E_c & -3I/L & 2I \\ -A & 0 & A.E_c & A & 0 & 0 \\ 0 & -3I/L^2 & -3I/L & 0 & 3I/L^2 & -3I/L \\ 0 & 3I/L & 2I & 0 & -3I/L & 4I \end{bmatrix} \quad \ldots(4.78)$$

Post multiplying Eq. (4.78) by the transpose of Eq. (4.76) we obtain the modified stiffness matrix Eq. (4.79) of a beam having the degrees of freedom shown in Fig. 4.23.



Orientation of degrees of freedom for an eccentric beam having the stiffness matrix given in (4.79)

Fig. 4.23

$$
\begin{bmatrix}
\frac{A.E}{2L} & 0 & \frac{-E_c.A.E}{2L} & \frac{-A.E}{2L} & 0 & \frac{E_c.A.E}{2.L} \\[2ex]
0 & \frac{3I.E}{2L^3} & \frac{3IE}{2L^2} & 0 & \frac{-3IE}{2L^3} & \frac{3IE}{2L} \\[2ex]
\frac{-E_c.A.E}{2L} & \frac{3IE}{2L^2} & \begin{matrix} E_c^2.A.E/2L \\ + \\ 2.I.E/L \end{matrix} & \frac{E_c.A.E}{2L} & \frac{-3IE}{2L^2} & \begin{matrix} -E_c^2.A.E/2L \\ + \\ I.E/L \end{matrix} \\[2ex]
\frac{-A.E}{2L} & 0 & \frac{E_c A.E}{2L} & \frac{A.E}{2L} & 0 & \frac{-E_c.A.E}{2L} \\[2ex]
0 & \frac{-3IE}{2L^3} & \frac{-3IE}{2L^2} & 0 & \frac{3IE}{2L^3} & \frac{-3IE}{2L^2} \\[2ex]
\frac{E_c.A.E}{2L} & \frac{3IE}{2L} & \begin{matrix} -E_c^2 AE/2L \\ + \\ IE/L \end{matrix} & \frac{-E_c AE}{2L} & \frac{-3IE}{2L^2} & \begin{matrix} E_c^2 AE/L \\ + \\ 2I.E/L \end{matrix}
\end{bmatrix}
\quad \ldots(4.79)
$$

Stiffness matrix for an eccentric beam of length 2L

5. Assembly of one panel's stiffness matrix.

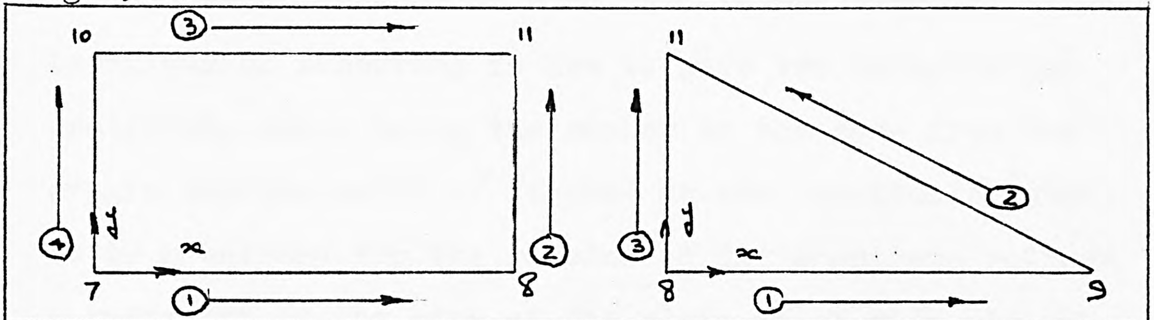
5.1 Numbering of degrees of freedom.

5.1.1 Global freedoms

As will be shown in chapter 7 any stiffness matrix maybe re-expressed in less rows and columns than its number of degrees of freedom. In this work, re-expressing is in terms of external and pseudo external degrees of freedom. To keep a check of the various freedoms an index of those global degrees of freedom must be prepared.

All the nodes of the structure which form corners of the panels are given by hand, positive integers which are unique to each individual corner node. The numbering of external degrees of freedom which are not pseudo external is completely automatic and is made by the subroutine FILE. The way FILE achieves this is to consider each corner in turn and then each side. Corner numbers of the current side are compared with the data held in core. If that corner has been previously processed then the reference number given to that corner is recovered and put into the record for this panel. If two adjacent corners have been processed then a search is made for the reference numbers allocated to the side between those two corners. If no numbers have been allocated then the next numbers in sequence are allocated.

Consider the two panels shown in Fig. 5.1, these would be defined as 7 8 11 10 for the rectangular panel and as the program expects four integers the triangular panel is defined as 8 9 11 0. When the program detects a zero as the fourth figure it immediately puts the variable NSH which is normally set at 4, to 3 indicating for the current panel's processing that the panel is three sided. The sides are numbered anticlockwise from the origin and the sequential numbering of the degrees of freedom along a side are in the directions indicated by the arrows of Fig. 5.1.



Side and corner numbering of panels.

Fig. 5.1

Considerable extra programing has been added at this stage to allow adjacent panels to have local axes at variance with those of adjacent panels. So that side 18 - 21 of the two panels shown in Fig. 5.2 will require

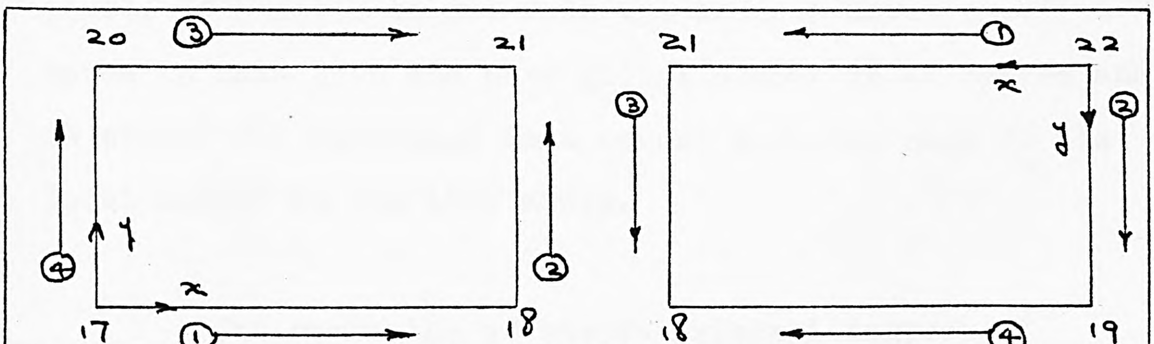


Fig. 5.2

reference numbers recovered from one panel to be reversed nodally but not individually before adding them to the records of the second panel. Had one edge been parallel to a local x axis and the other to a local y axis then the reference numbers would also require to be rephased.

It is necessary define by hand, which degrees of freedom are restraints. These may be external and/or internal. This was originally just the number of degrees of freedom from the origin counted in the way local numbers are allocated. This proved prone to error as the number of degrees of freedom at each node is not constant. The technique of numbering is now to give two integers per restraint, these being the number of the node from the origin and the order of freedom at that particular node. It is necessary for the routine to differentiate between a restraint on the edge of the plate which will already have been allocated its global reference number and an internal which does not yet have a global reference number. For spring ends a similar numbering system applies. Both the spring endings and restraints eventually convert the two supplied numbers to an ordered local number which is compared with the index of all local numbers of that panel, if a match is not made the process moves on. If a match is made then the next global number is allocated and is stored for reference in a vacant location next to its local number in the list above.

The numbering of pseudo external degrees of freedom on a particular panel is done near the beginning of subroutine ASSEMB's processing of that panel. They are numbered sequentially from the greatest number given to any external degree of freedom in the whole structure.

5.1.2 Local freedoms.

A second index is kept for assembly purposes. This index is an ordered list of all the degrees of freedom on the plate irrespective of their position on that plate. The numbering is a simple progression in the order that the degrees of freedom will be assembled. As before they will be grouped in blocks of six locations per node. In all cases a standard convention is adopted for the storage of these reference numbers.

The numbering of freedoms at a node whether for global or local indices are as shown in Fig. 5.3

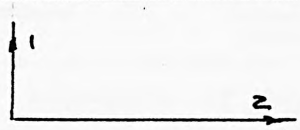
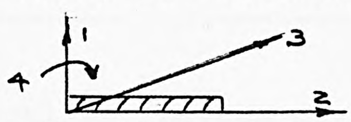
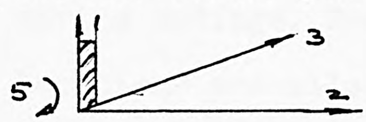


Plate node, no stiffener present.



Node with stiffener present in x direction.



Node with stiffener present in y direction.

1	2	3	4	5	6

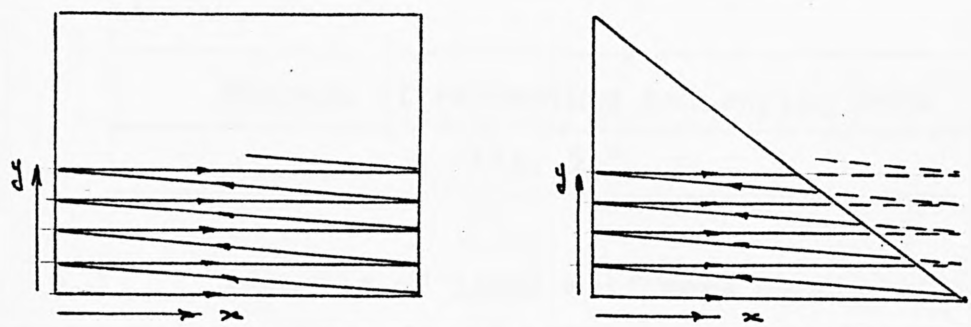
Representing the six locations in array Idata used for the storage of reference numbers of degrees of freedom for this node.

Location.	Freedom stored.
1	Local y
2	Local x
3	Lateral z direction
4	Rotational about local y
5	Rotational about local x
6	Always void

Scheme adopted for numbering degrees of freedom.
Fig. 5.3

5.2 Indices.

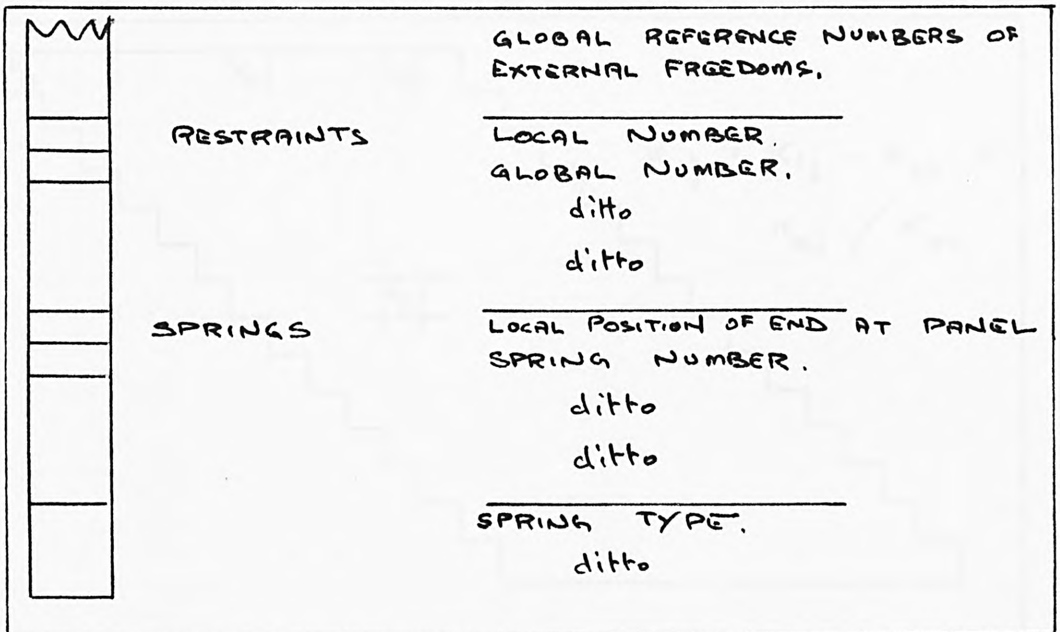
5.2.1 There are two main indices temporarily stored for each panel. As mentioned in chapter 3, one index contains the local numbering and the other the global numbering of degrees of freedom at edge nodes plus certain declared internal freedoms. The order in which these freedoms are stored is shown in Fig. 5.4.



Order of assembly of degrees of freedom

Fig. 5.4

Two smaller indices may exist one for restraints, the other for spring endings. For the restraint index twice the number of locations are allocated. This is so that local and global reference numbers can be stored side by side for ease of recovery. For the spring index the number of locations required is three times the number of springs. Two grouped numbers are for the freedom and spring reference numbers, the third location is to store the type of spring attached to that freedom. The storage locations of the reference numbers of the global and local degrees of freedom are shown in Fig. 3.8 and Fig. 3.12. The storage of the restraints and spring data is as shown in Fig. 5.5.

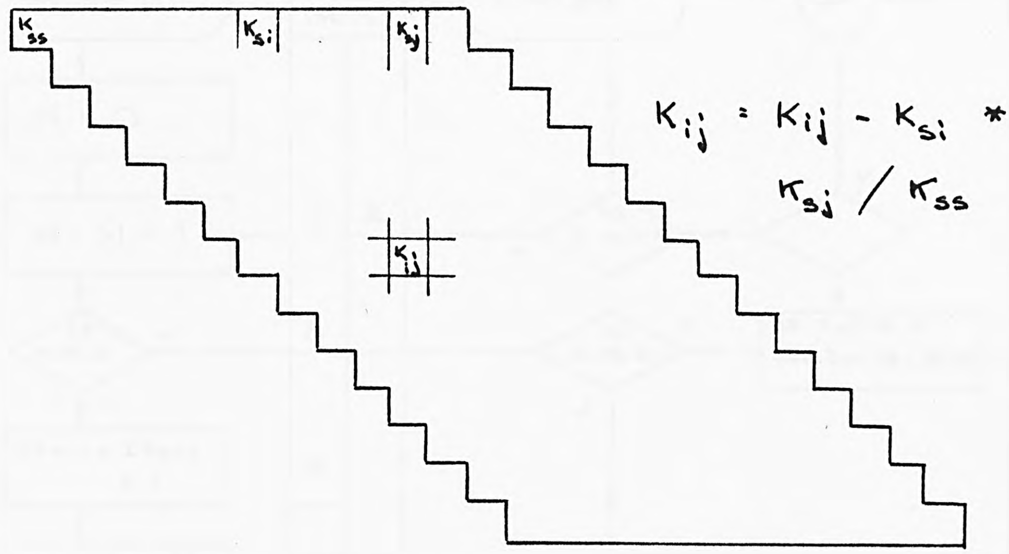


Storage of restraints and spring data

Fig. 5.5

5.3 Assembly of local stiffness matrix.

5.3.1 On the face of it assembling a stiffness matrix by overlaying various square matrices seems a little cumbersome and that building up a final matrix row by row would be a more logical method. In reality to obtain a full stiffness matrix by adding individual whole matrices is the easiest way. But consider the case where it is arranged for some of the rows of the stiffness matrix not to appear in the final matrix. If a row by row approach is not adopted as in the usual wave front method we get a matrix as in Fig. 5.6. This shows the space (which may not be available) which has to be occupied before the s^{th} row can be eliminated. If now the next row to be formed is put into the newly vacated site then considerable and complex accounting must be maintained to keep track of where rows are being inserted.

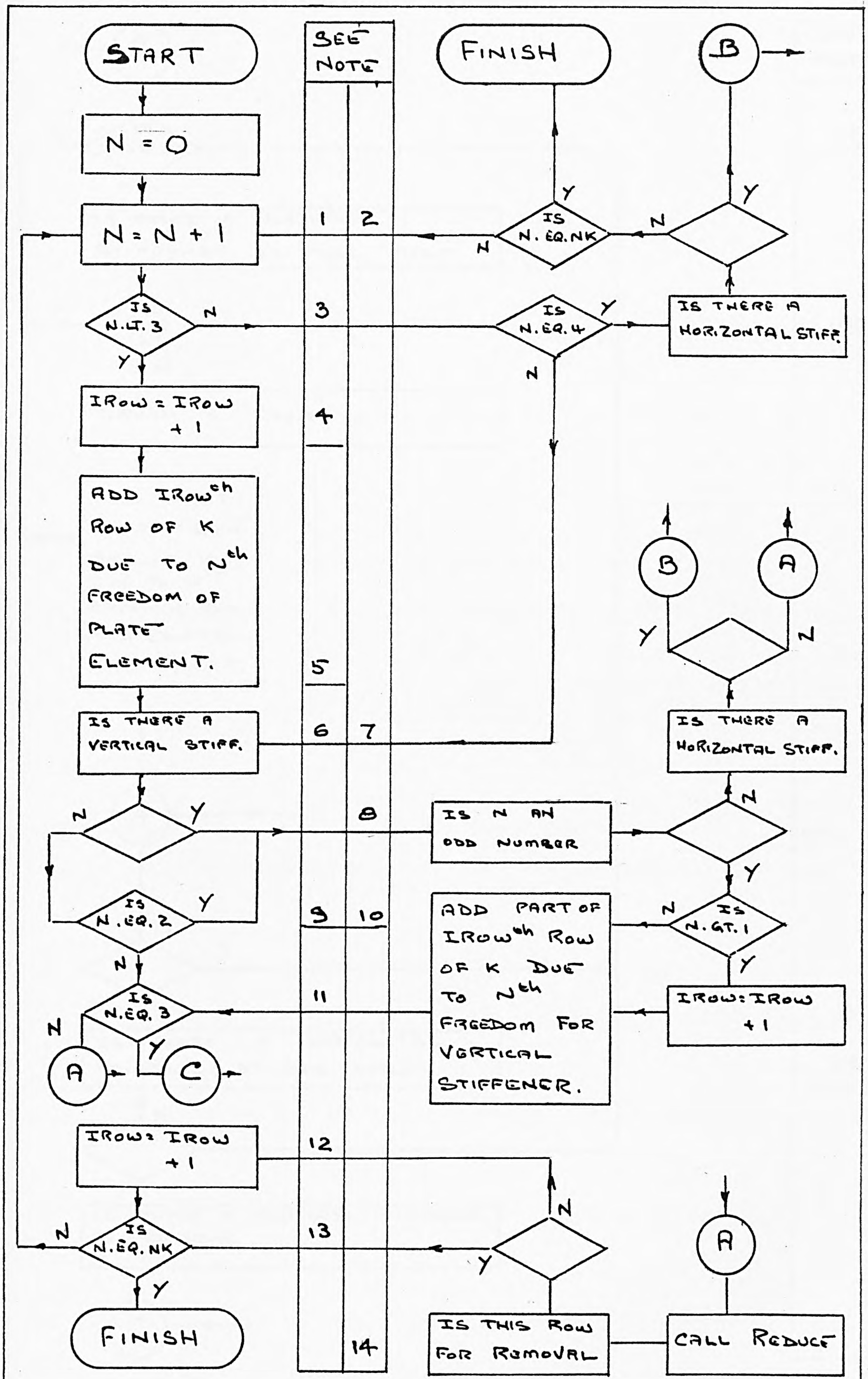


Modification to K_{ij} term necessary as a result of removing the s^{th} row

Fig. 5.6

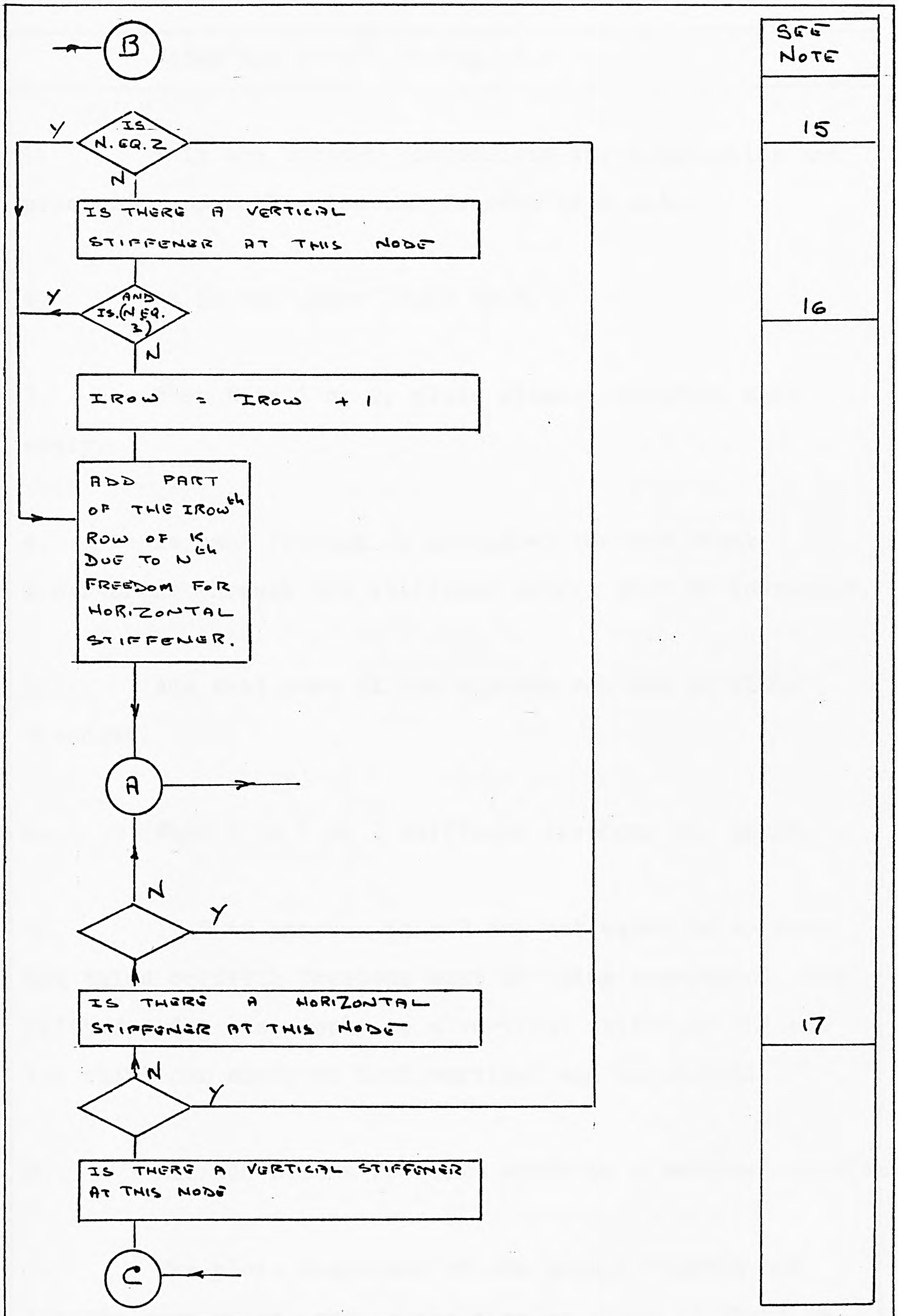
5.3.2 If rows are to be removed individually then it is suggested that individual row assembly is advantageous. The flow chart at Fig. 5.7 shows how to assemble row by row and the table given in Fig. 5.8 gives values for the vector to be used in the case of a plate element.

By removing a row we are trying to remove all references to a particular displacement, but other rows will contain coefficients related to the displacement being removed. To keep the overall number of equations and unknowns equal it is necessary to remove these other coefficients. In the case of a row being removed from the matrix assembled row by row, some of the other rows having coefficients to be removed have themselves already been removed whilst others have not been formed. The flow chart



Flow chart for the method of assembling individual rows at a single node (Part 1).

Fig. 5.7(a)



Flow chart for the method of assembling individual rows at a single node (Part 2).

Fig. 5.7(b)

Notes applicable to Fig. 5.7

1. N is the counter controlling the examination and processing of each potential freedom at a node.

2. NK is the upper limit of N.

3. When N is 1 or 2, plate element freedoms must apply.

4. As each freedom is processed the row count progressing through the stiffness matrix must be increased.

5. Add that part of the current row due to plate freedoms.

6. When N is 1 or 2 stiffneer freedoms may apply.

7. If N is greater than 2 and not equal to 4, then the third or fifth freedoms must be being considered. The fifth freedom must apply to a vertical stiffener whilst the third can apply to both vertical and horizontal.

8. All odd number freedoms apply to a vertical stiffener.

9. The plate component of the second freedom has already been added, now is the time to check if there is a horizontal stiffener present and its contribution.

10. To have got this far shows that a vertical stiffener exists. If N is still at 1 then action for the

plate's first freedom will have incremented the row count. For the other two freedoms the row count must be increased here.

11. There still exists the chance of an additional segment to this row when it is for the third freedom from a horizontal stiffener.

12. Once N is greater than 2 the row increment at 4 above is not in the remaining path but is still needed at some points.

13. At the end of each loop the question is asked "Is this the final loop?".

14. Take the necessary action to modify and remove coefficients.

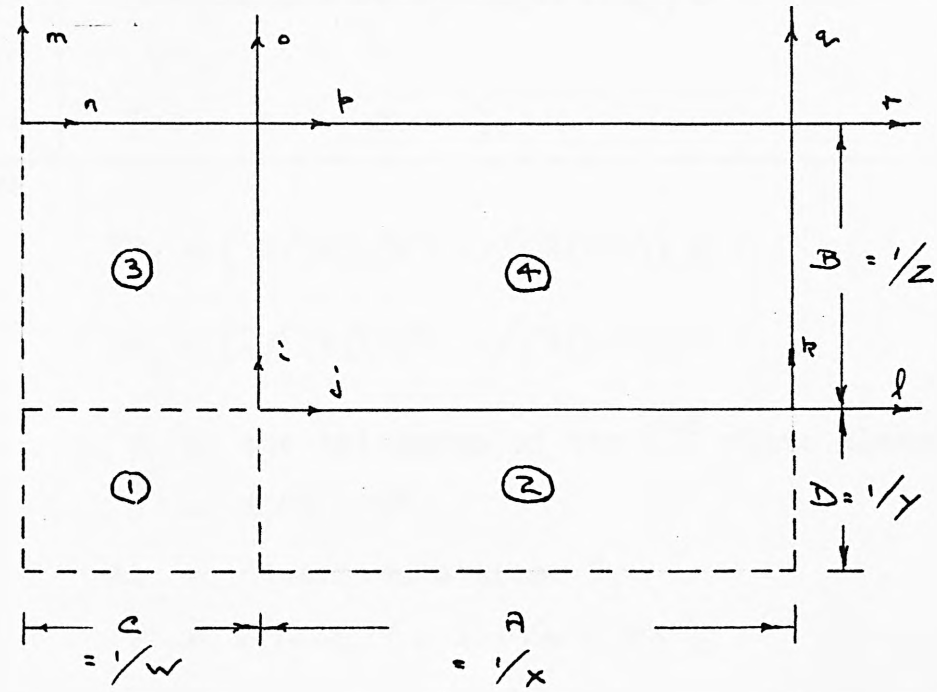
15. If N is equal to 2 then only a horizontal stiffener's contribution needs to be added.

16. If N is equal to 3 and there is a vertical stiffener present then the row count will have already been dealt with by that vertical stiffener sharing the third freedom with the current horizontal stiffener.

17. If there is neither a horizontal nor vertical stiffener then the reduction process may start.

End of notes for Fig. 5.7

at Fig. 5.9 shows how this trouble can be overcome.



Location in vector	Contents at location
i,i	$(F(Z(A4+C3)+Y(A2+C1))+X(B4+D2)+W(B3+D1))U$
i,j	$VP_1 - VP_2 - VP_3 + VP_4$
i,k	$(FF(Z.A4+Y.A2) - X(B4+D2))U$
i,l	$VM_2 - VM_4$
i,m	$(-C3.Z.FF - B3.W)U$
i,n	VP_3
i,o	$(B3.W + B4.X - (C3+A4).Z.F)U$
i,p	$VM_4 - VM_3$
i,q	$(-A4.Z.FF - B4.X)U$
i,r	$-VP_4$
j,j	$(F(X(B4+D2)+W(B3+D1))+Z(A4+C3)+Y(A2+C1))U$
j,k	$-VM_2 + VM_4$
j,l	$(-F.X(B4+D2) + A2.Y + A4.Z)U$
j,m	VP_3
j,n	$(-FF.B3.W - C3.Z)U$

j,o	$VM_3 - VM_4$
j,p	$(FF(B3.W + B4.X) - Z(A4 + C3))\,U$
j,q	$-VP_4$
j,r	$(FF.X.B4 + Z.A4).-U$

where

$$VP_i = (\nu/(4(1-\nu^2) + 1/(8(1+\nu)).E.T_i$$

$$VM_i = (\nu/(4(1-\nu^2) - 1/(8(1+\nu)).E.T_i$$

T_i is the thickness of the i^{th} plate element.

$U = E/8(1+\nu)$

Ai = dimension A times T_i

$F = 8(1+\nu)(1/12 + 1/4(1-\nu^2))$

$FF = 8(1+\nu)(1/12 - 1/4(1-\nu^2))$

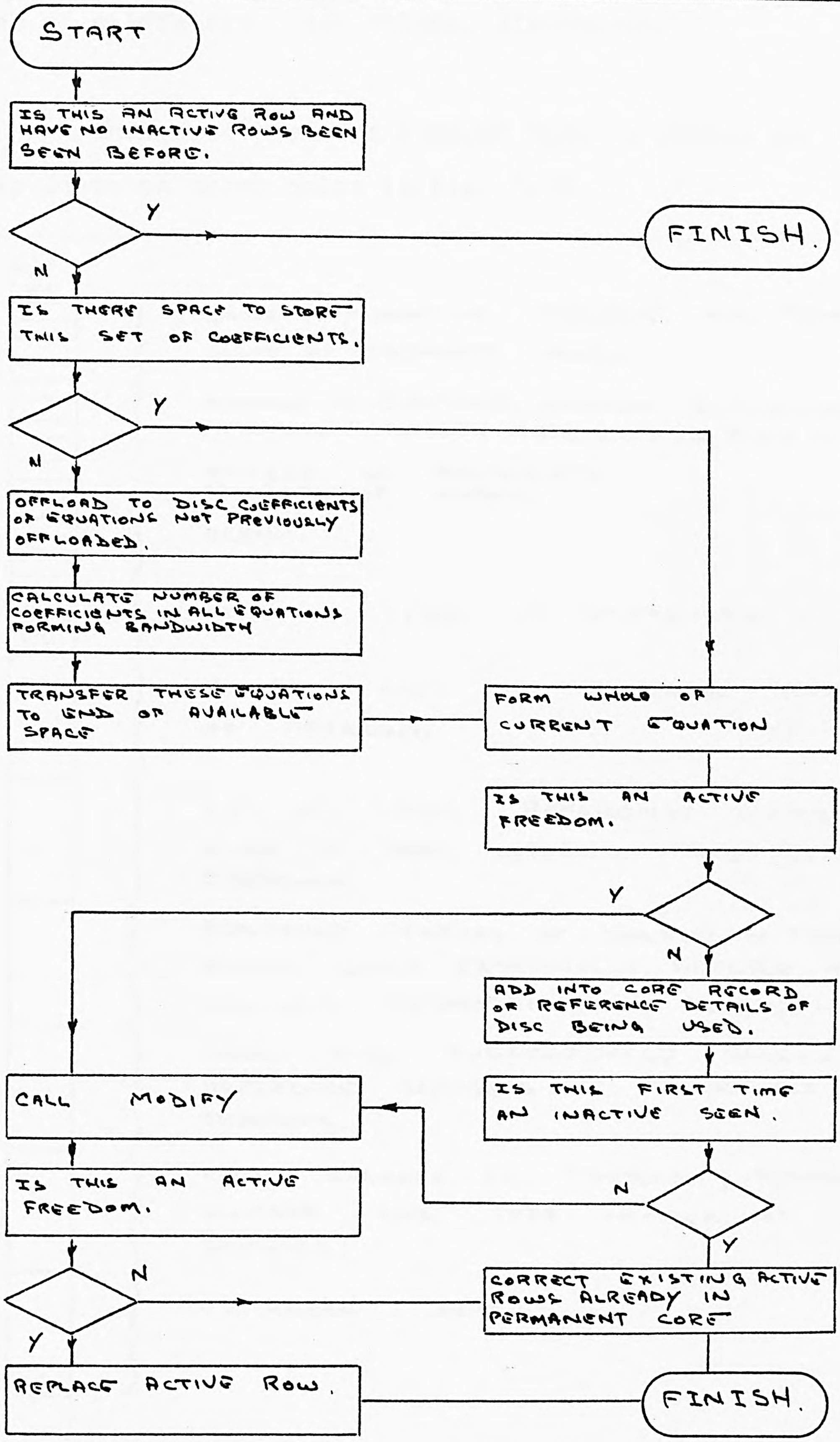
Table showing the make up of the stiffness vectors for the i^{th} and j^{th} freedoms of a panel as shown.

Fig. 5.8

5.3.3 The process starts with the first degree of freedom of the first node and proceeds node by node moving onto the next node as triggered by the presence or absence of any reference number in the index of local numbering. The stages of the process are as follows.

5.3.3.1 The following stages are done once per panel.

a. Recover from disc details required by the program to assemble the local stiffness matrix such as corner numbers(which allows a decision to be made as to whether this is a triangular or rectangular element), reference numbers of degrees of freedom, restraints, springs,

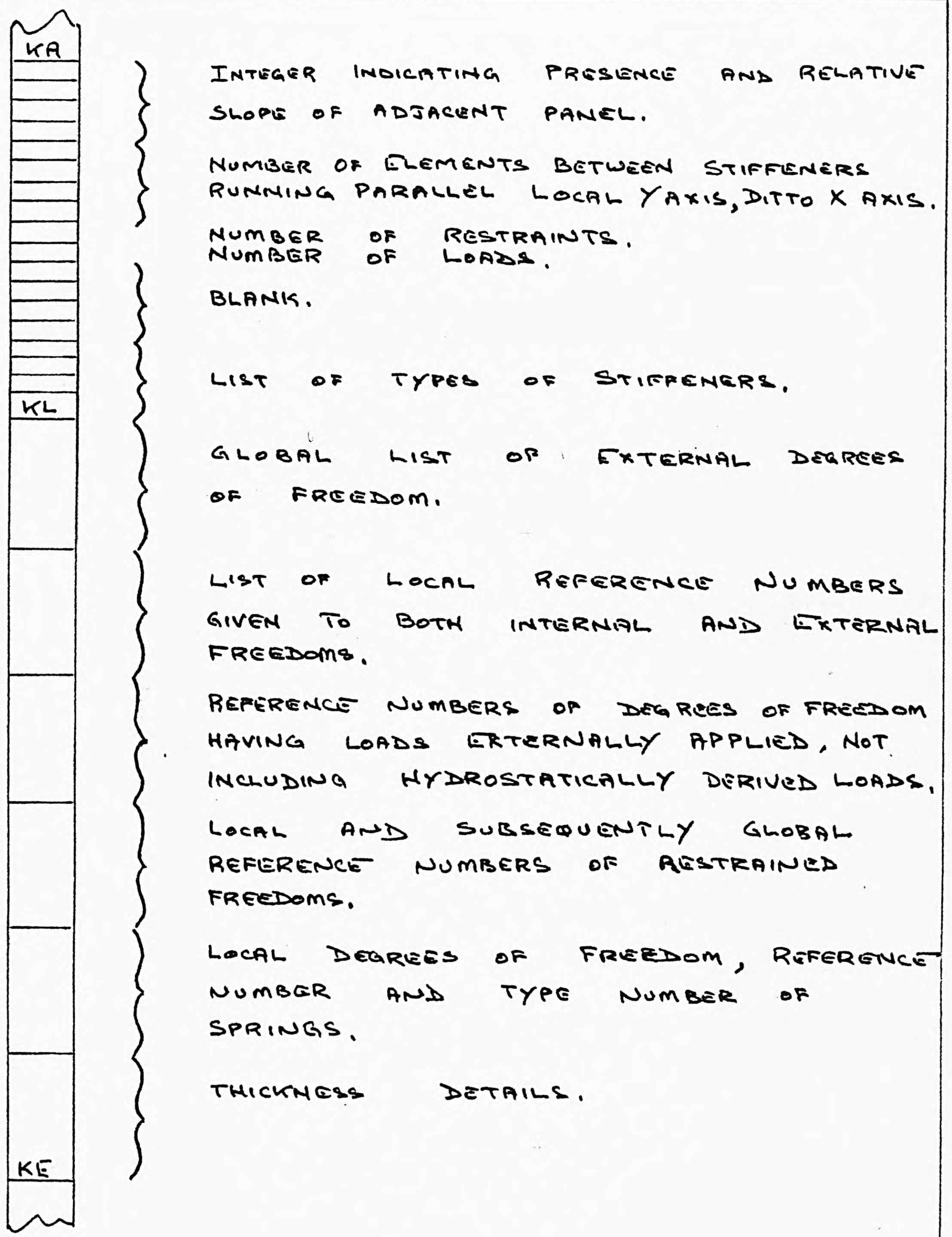


Flow chart showing the outline for removing coefficients of inactive freedoms.

Fig. 5.9

types of stiffeners, load values, dimensions.

When recovered the integer data is stored in array Idata as shown below in Fig. 5.10



The location of integer data in array Idata and space occupied by each group at the start of sbr. ASSEMB.

Fig. 5.10

Similarly real number information recovered from disc is stored in array Real as shown in Fig. 5.11. This data overwrites previous panel information.

DIRECTION COSINES OF CURRENT PANEL.	9
ELEMENT SIZE IN LOCAL x DIRECTION.	NSTY+1
ELEMENT SIZE IN LOCAL Y DIRECTION	NSTX+1
BLANK	3
DETAILS OF THICKNESSES APPLYING TO CURRENT PANEL.	NT
LOAD VALUE AT EACH FREEDOM UNLESS THERE IS NO APPLIED LOAD	NILOAD.

The location of real number data in array Real and space occupied by each group at the start of sbr. ASSEMB.

Fig. 5.11

In the locations before the variable KB of array Real, stiffener details and co-ordinates of all corners of the structure are permanently stored.

The position on disc for the start of the information shown in Fig. 5.10 is found in the permanent index held in Idata at Idata(L) for the L^{th} panel. The real data for Fig. 5.11 can be located by reference to Idata(L+NPANEL+1).

b. Modify record of corners to indicate their activity state. During the loading of panel data a count is kept of the number of times each corner is mentioned during panel definition. During assembling of local stiffness matrices this value is reduced by one for each of the current panel's corners. This enables a decision to be made on the activity state of degrees of freedom along a side. When the corner appearance value of a particular corner reaches zero then all the sides having that corner as one end become inactive and their sign in the index monitoring the activity status changes to a negative.

c. Calculate the bandwidth of the current local stiffness matrix. This is undertaken in two parts. First a set of triggers is set up which indicate the presence of stiffeners with only one plate element between them, the presence of edge stiffeners and a combination of both of these. Secondly using these triggers form the value of the bandwidth from the relationship

$$\text{Bandwidth} = \text{ISX} * \text{N} + \text{NSTY} * \text{M} + \text{NB}$$

where N,M and NB are arrived at from various combinations of the triggers. N is usually 2 but is increased to 4 if there is a stiffener running in the local x direction. M takes account of intersecting stiffeners and is 2 if there are stiffeners in the y direction but none in the x direction and becomes 1 if there are stiffeners in both directions. NB adds on the number due to overlapping at the end of a row with the adjacent row.

d. Form a new index of global reference numbers of degrees of freedom. This is necessary so that internal restraints and spring endings can be recorded in their correct positions relative to externals already processed previously in subroutine FILE. Opportunity is taken as panel information is scanned to update counts of degrees of freedom for these particular internal values and to change signs in the indices. This new index will be called the KE index. The index of the global numbers previously shown in Fig. 5.10 will be called the KL index. Reference has been made to negative values in indices, these are now summarized.

A negative in the KE index indicates an internal restraint and will appear in the KL index as positive. A negative in the KL index indicates a degree of freedom which does not exist for the current panel in its local stiffness assembly, but will appear when the terms for that node are brought into alignment with global axes. A positive value in KE index but not appearing in KL index indicates a spring ending.

e. Check that there is sufficient space left in array Real to store the current panel's matrix.

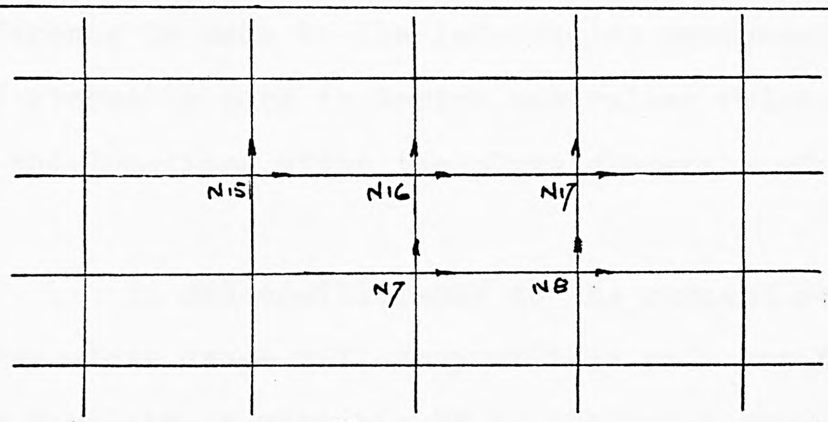
5.3.3.3 The following stages are done once per node.

a. Calculate the dimensions required. In this case there are four dimensions but only two have to be recovered from array Real as the other two are the two previously recovered from Real for the last element.

b. Find the value of the thickness for the current plate element. Thickness details are stored in Real straight after the spring details. The first value of T is recovered and the repeat number found in the array Idata. Each time an element is seen the repeat number is reduced by one. When the repeat number gets to zero a new thickness value is recovered. When the repeat number is found to be negative the value of the thickness then in use is kept for the remainder of that panel and no attempt is made to change it.

c. Question. Is there a vertical stiffener present and is there also a horizontal stiffener present? Their presence is easily seen by examining if the plate element counter is set at one. The counter could be at one on a front edge, in this case an additional check is run to see if there is an edge stiffener there.

d. Load the arrays used to guide the assembly of the row of the stiffness matrix into the right longitudinal positions. The values to be put into the matrix are stored in array Real. The question is how to get the value into the correct location. For a plate element we have for freedom i_1 ten coefficients. These are stored in Real(LIMIT + 1) to Real(LIMIT + 10). Examination of the index storing local reference numbers which are stored immediately after the KL index will give a guidance as to where these coefficients go. An array Num10 is used to store the information retrieved from the local index. Num10 (1), (2),(3) and (4) are given the first two integers of the six locations of the node at position N7 in Fig. 5.12 and the first two integers of the next group of six of the local index, respectively. To load the remaining six locations of NUM10 the counter looking through the local index is increased by ISX-1 and the next first two integers for the next three nodes are added.



N_k is the k^{th} node.

i_1 ↑ is the first order freedom at that node.

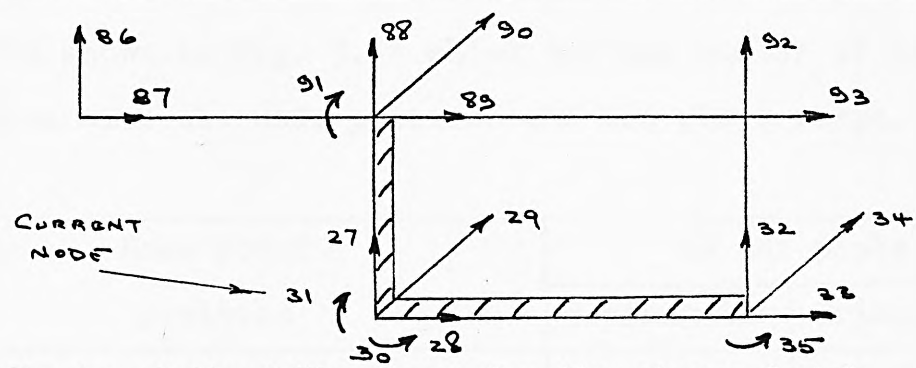
i_2 → is the second order freedom at that node.

Fig. 5.12

Some adjustment to the contents of NUM10 is necessary if the node under consideration is on an edge. If this is a left hand edge the contents of the 5^{th} and 6^{th} locations (the freedoms at node N15 of Fig. 5.12) are meaningless. In such a case the contents of the 7^{th}, 8^{th}, 9^{th} and 10^{th} locations are moved to the left by two places to fill those locations. Similarly on the right hand edge the 3^{rd}, 4^{th}, 9^{th} and 10^{th} locations' contents are meaningless and there is a movement of figures to the left. Two other arrays exist NUM6 and NUM12. They do respectively for vertical and horizontal stiffeners what NUM10 does for plate nodes. Fig. 5.13 shows the storage positions of the contents of the NUM arrays.

e. Formation of beam stiffness matrix. This is started at each node at which there is a stiffener, if the stiffener is in all respects as the last one then no further action is taken. If new values are needed then reference is made to the information recovered from disc and stored in core to derive new values which are stored in the locations after the plate element coefficients.

f. Add coefficients to the current row of the array. This stage will stay at this node until all rows are complete. A variable NK is set up to govern how many loops exist. This will be 2 unless a stiffener exists.



Contents of the NUM arrays for the above sketch.

Num10(27 28 32 33 86 87 88 89 92 93)

NUM6 (27 29 31 88 90 91)

NUM12(28 29 30 33 34 35)

Positions of coefficients in array Real as indicated by the storages in the NUM arrays.

Load	27	28	29	30	31	32	33	34	35

86	87	88	89	90	91	92	93

Fig. 5.13

As assembly progresses a count of the local degrees of freedom is kept by the variable LDF, as a count of rows by the variable IROW will be of no use for such things as scanning array Real for load values stored by local reference numbers. The outline of the assembly routine at a node is shown in the flow chart at Fig. 5.7. The first location of a row of coėfficients to be added is the term for the load vector. The load vector is not assembled as a separate vector but for ease of manipulation and a small saving in space is intermingled with the stiffness matrix. The control of the positioning of coefficients is by the three NUM arrays described in d above. The vertical freedoms at a plate node will normally have 10 coefficients and the horizontal 9. A variable KN is set to the number of coefficients to be loaded. The table shown in Fig. 5.14 shows how the number of coefficients varies with the node position and the plate shape.

Node point position	KN for plate	
	Rectangular	Triangular
Top left hand corner	4	2
Top right hand corner	2	N/A
Elsewhere on the right hand side	6	4
Elsewhere on the left hand side	8	8
Elsewhere on the top	4	N/A
One node before the right hand edge when the node also falls on the left hand edge.	N/A	6
Elsewhere	10	10

Fig. 5.14

As the combined stiffness matrix and load vector is loaded into a single one dimensional array, a variable IRW is used to keep a count of all locations used. Because of the removal of non external degrees of freedom there is a mismatch between the consecutive number of the row being loaded and the consecutive number of the row into which it is going. So that if the first integer in array NUM10 is N1, then conventionally this would guide the first coefficient of the row vector into the leading diagonal position (N1,N1). In this case the row may be the $(N1 - N2)^{th}$. To obtain compatibility, the three NUM arrays are modified by a variable KP so that Real(IT + NUM10(K + KP) + IRW - NUM10(1 + KP) + 2) is the location for the plate vector coefficient Real(LIMIT + K + KP *10), where K is the loop counter for loading the N^{th} stiffness vector of the current node. K loops from 1 to KMAX. The table at Fig. 5.15 shows these various values.

5.4 Storage of the local stiffness matrix.

5.4.1 Outline.

5.4.1.1 There are two stiffness matrices to store in the array Real. One is the stiffness matrix of the wavefront existing before the current panel was added. The second is the storage of the current panel's stiffness matrix expressed in terms of external and pseudo external degrees of freedom. Also to be stored in the array Real are details of the next stiffness vector for the next assembly stage of the current panel's matrix and similarly

Element type.	N	KP	Maximum value of K.	Location in array Real for coefficient.	Location of Real coefficient to be transferred.
Plate	1	0	KN - KP	IT + NUM10(K + KP) + IRW	LIMIT + K + KP * 10
	2	1		-NUM10(1 + KP) + 2	
Horizontal stiffener	1	N/A	If on right	IT + NUM12(K + N - 2) + 2 + IRW	LIMIT + 19 + KP + K
	2	0	edge of plate	- NUM12(N - 1) - KCOR	
	3	6	5 - N	where KCOR = 0 unless N = 3	
	4	11	Elsewhere	when KCOR = 1	
	5	N/A	8 - N		
Vertical stiffener	1	0	If on top	IT + NUM6(K + (N - 1)/2) + IRW	LIMIT + 34 + KP + K
	2	N/A	edge of plate	- NUM6(1 + (N - 1)/2) + 2	
	3	6	3 - (N - 1)/2		
	4	N/A	Elsewhere		
	5	11	6 - (N - 1)/2		

Table showing values of variables used to correctly locate stiffness matrix coefficients.

Fig. 5.15

when applicable the two stiffness matrices due to the orthogonal stiffeners.

5.4.1.2 This information is, other than that for the two stiffness matrices only transitory and is therefore stored at the far end of the array after the variable LIMIT allowing the existing wavefront matrix to be stored at the beginning and runs through to the location given the variable name IT. The current panel's matrix is then stored between IT and LIMIT. This arrangement is shown in Fig. 7.9. To minimise calls to disc, coefficients to be offloaded to disc are collected at the end of the known empty space at the variable location LIMIT, subsequent coefficients are then stored to the left of the last as coefficients for retention are stored to the right from the KB end. When the two sets of coefficients meet or there is insufficient space left between them to store a full set, then those coefficients due for offloading are thus processed. The disc record system maintained in array Idata is kept continually updated as if the coefficients were being put straight to disc without any intermediate storage. This will be explained in more detail in chapter 7.

6. Loads and Restraints.

6.1 Availability of loading conditions.

6.1.1 This package allows the structure to be acted upon by any combination of the following:

1. Individual translation point loads.
2. Individual rotational point loads.
3. Hydrostatic head subjected to longitudinal acceleration relative to the structure.
4. Hydrostatic head subjected to transverse acceleration relative to the structure.
5. Hydrostatic head subjected to vertical acceleration relative to the structure.
6. The structure's own weight whilst it is being subjected to longitudinal acceleration.
7. The structure's own weight whilst it is being subjected to transverse acceleration.
8. The structure's own weight whilst it is being subjected to vertical acceleration.
9. Thermal loads.

6.2 Availability of restraints.

6.2.1 This package allows the structure to be wholly or partially restrained from deflecting as below:

1. The total restraint of any nominated

degrees of freedom.

2. The partial restraint of any nominated degree of freedom. This is achieved by springs, whose stiffnesses may be declared in the data or automatically formed, in which case they will be stiff in comparison with the structure.

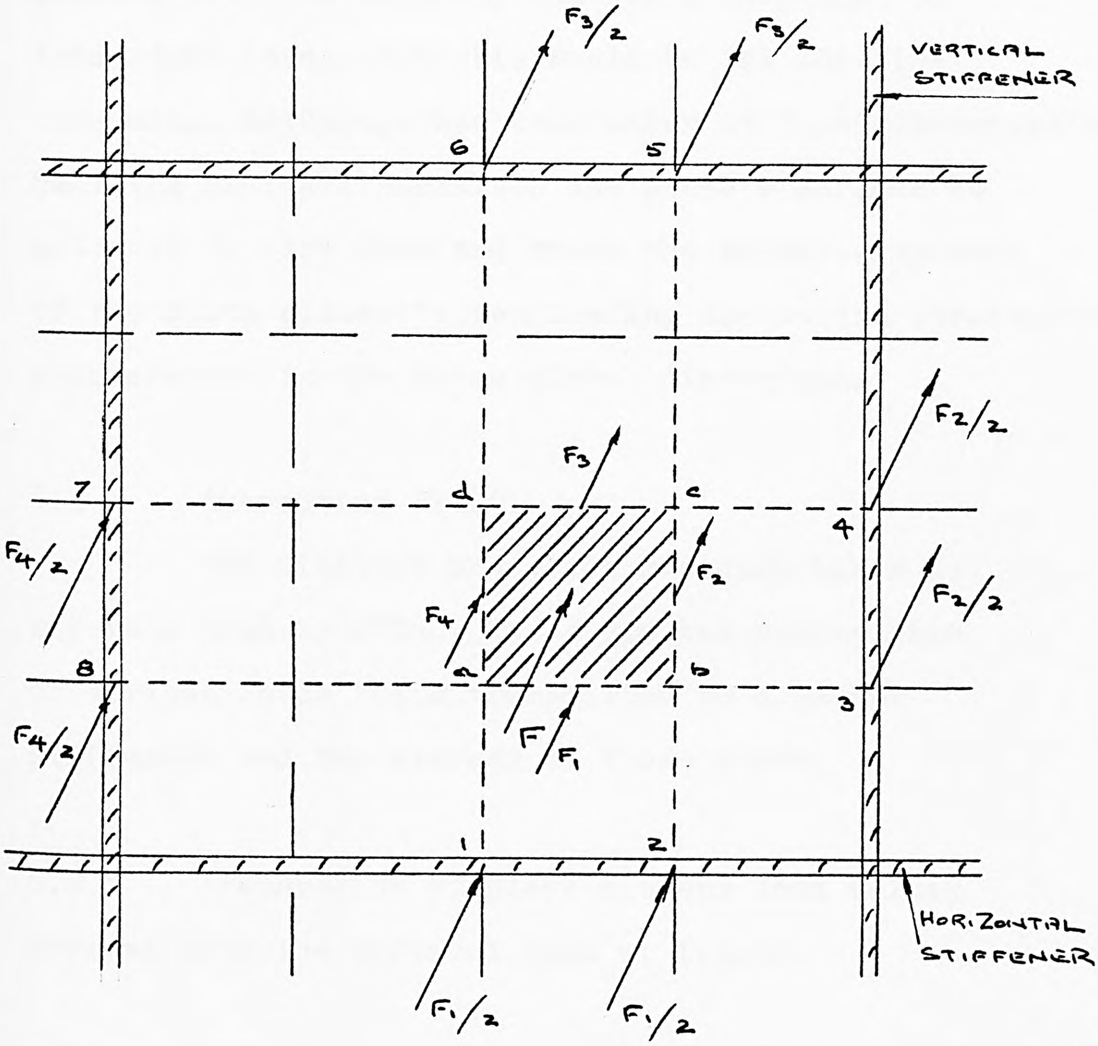
6.3 Point load storage.

6.3.1 Individual external loads may be applied at any node containing a degree of freedom compatible with that load. The values of externally applied loads together with those components of structural weight occuring normal to the plate are stored consecutively in array Real immediately after details of the thickness values as shown in Fig. 3.12. To minimise storage space, locations with no load are ignored, this though causes a load index to be required to be maintained in array Idata. On balance there is an overall saving in core space by keeping an index to take account of the lack of void spaces in the list of loads. The formation of the panel's load vector is progressed at the same time as the assembly of the panel's stiffness matrix. As previously explained the stiffness matrix is assembled row by row. At the formation of any row, the next unused value of the load index is compared with the local freedom then being assembled. If the values are equal then the value of that load is recovered from its segment of Real and transferred to the location immediately before the leading diagonal term of the current row as shown in Fig. 7.10

6.3.2 The index value of a load is one plus the number of degrees of freedom from the local origin which have to be examined before making it the current value. It was found that numbering in this style by hand was prone to error. The technique used now for data on such translational or rotational loads is to read in for each load the local node number at which the load is being applied followed by its order at that node. This requires a small loop in the load storage routine which scans the local index of all the freedoms of the panel until the block of the specified node is found, the value in the appropriate location for the load's freedom is then put into the load index. If an external load is known in global terms and the direction cosines of the local axes are not equal to those of the global axes, then the value of that load must be resolved into the directions of the local axes before being put into the data.

6.4 Hydrostatic loads.

6.4.1 Hydrostatic loads may be automatically read in by defining the head and density of the liquid. Since there are third order translation freedoms locally available only where there is a stiffener, hydrostatic loads must be calculated for each individual plate element and then shared out between the adjacent stiffeners supporting that element. It can be seen from Fig. 6.1 that the hydrostatic load on element abcd is taken by the stiffeners' out of plane freedoms at 1,2,3,4,5,6,7 and 8.



Assumed locations for the distribution of hydrostatic loading from a single element.

Fig 6.1

As the solution moves on through the panel some of the storage spaces will be used more than once. At the same time any change in the vertical component of the element's position will cause a change of applied head and hence the applied loading. Subroutine HYDRO has been written to deal with the calculation of the hydrostatic loads normal to the panel and the updating of storage locations. The hydrostatic loading could be added by hand as a long list of individual loads, but this would be far too time consuming. Advantage has been taken of this subroutine's handling of loads normal to the plate's surface to allow it to also form and store the normal component of the plate element's own loading due to the structure's acceleration in the three global directions.

6.5 Subroutine HYDRO.

Two distinct processes are undertaken at the same time by HYDRO. These are the preparation of various loads being transmitted to adjacent stiffeners and the storage of those loads.

6.5.1 Preparation of plate element load values derived from the vertical head of liquid.

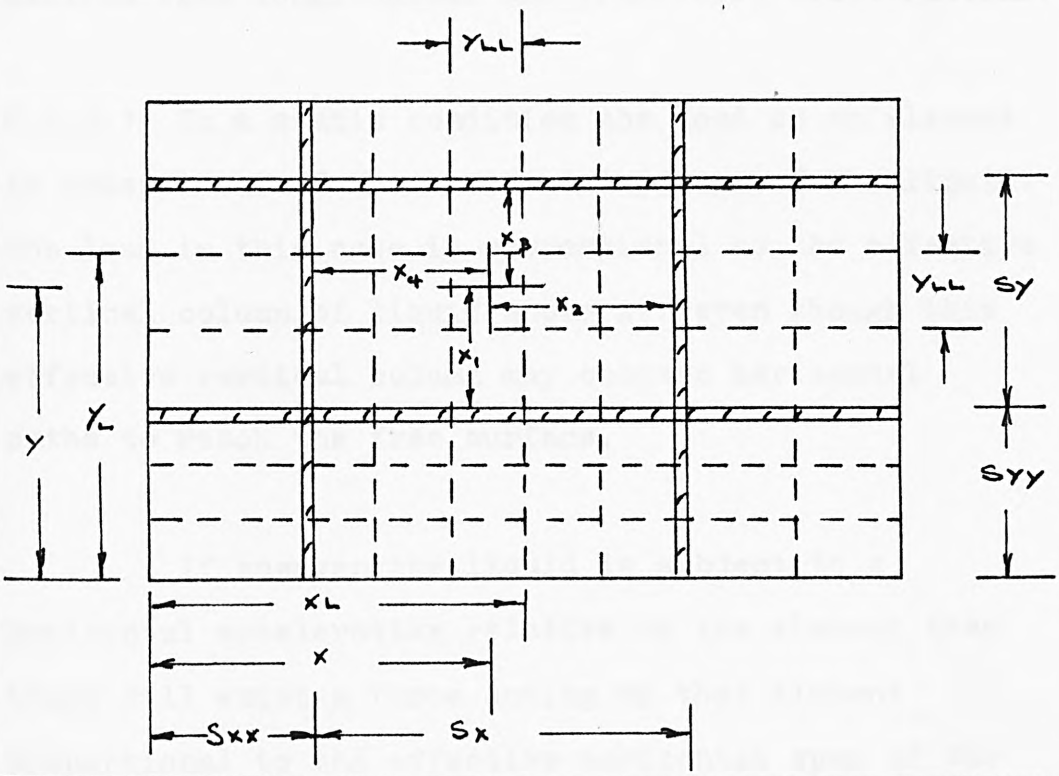
At the start of each panel's call to HYDRO values are derived for:

1. The change of global head due to unit change of position parallel to the local x axis.

2. The change of global head due to unit change of position parallel to the local y axis.

3. The head applicable at the panel's origin.

6.5.1.1 The value of the hydrostatic pressure on an element is a function of the head applied to that element's centre of pressure. In this package it is assumed that the centre of pressure and the centroid are coincident. Each panel is scanned in the technique used for assembly, by starting at the origin and moving up the local y axis to the next row after each row has been examined from the y axis working outwards. Counters are maintained giving the local co-ordinates of the bottom left hand corner of the current element relative to the panel's origin, to which are added the respective half lengths of the sides of the element.



Variable names given to positions used in updating dimensions relating elements to their surrounds.

Fig. 6.2

6.5.1.2 By multiplying the local co-ordinates of the centroid by the appropriate change of head factor a value can be obtained to subtract from the head at the origin, to give the head applied to the current plate element. The load being applied normal to that element due to the vertical head can be found by multiplying together the derived pressure and the element's area. This value can in turn be multiplied by any acceleration/gravity ratio that may be induced by heaving.

6.5.2 Preparation of plate element load values derived from longitudinal and transverse accelerations.

6.5.2.1 In a static condition the load on an element is independent of the horizontal extent of the liquid. The load in this case is proportional to the effective vertical column of liquid above it, even though this effective vertical column may contain horizontal paths to reach the free surface.

If however the liquid is subject to a horizontal acceleration relative to the element then there will exist a force acting on that element proportional to the effective horizontal span of the liquid acting on the element.

6.5.2.2 In the case of a vertical head, the value of the applied head is being continually updated to take account of the vertical position of the element. This is necessary because the head is varying over a wide range and the use of a mean value would introduce considerable inaccuracy. For a horizontal head caused by a transverse or longitudinal acceleration a mean value is used. In the majority of cases the panels are vertical or horizontal and so the mean value is the true value. In those cases where it is felt that the errors introduced from using the mean head are not acceptable, accuracy my be improved by representing the sloping panels in more than one part, with each part having its own head value.

6.5.2.3 Since only mean horizontal heads are used the calculation of the total load coming onto a plate is simplified in that only one head value has to be continually updated. On going to the next plate element it is quite easy to calculate the load being applied from the horizontal heads and sum them with the load from the vertical head already stored.

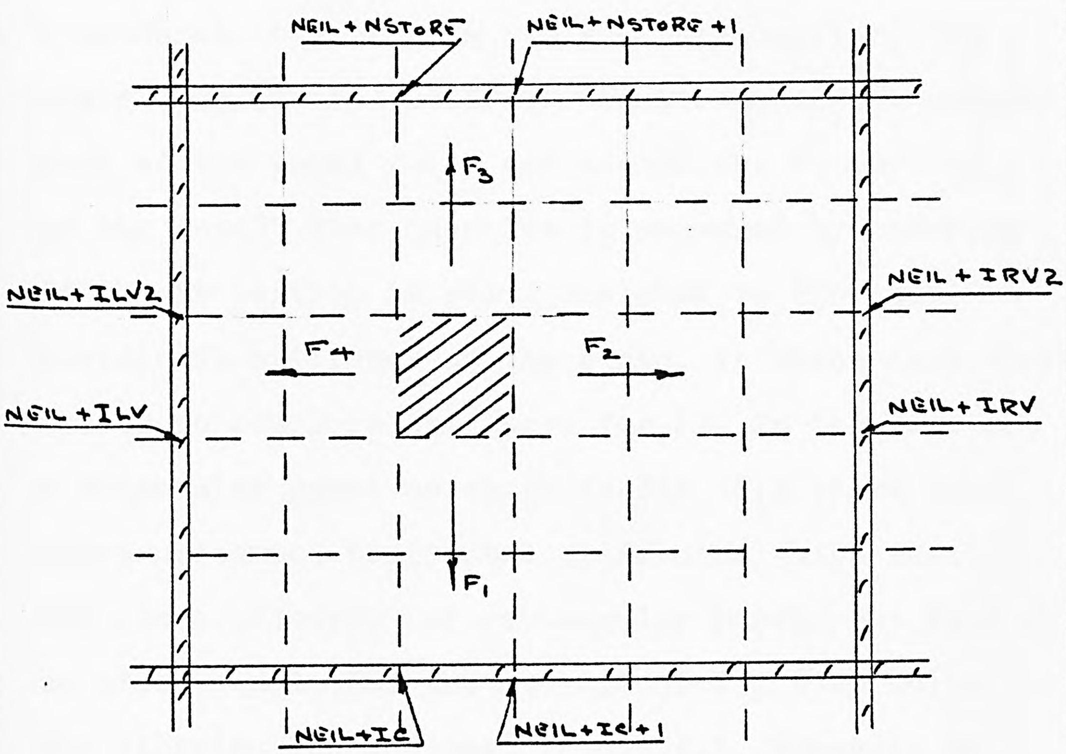
6.5.3 Dynamic loadings.

Reference has just been to gravity loads due to ship-motion. In Ref.(34) a method is outlined to take account of ship-motion when obtaining sea-caused loads. The resulting dynamic pressure values are converted to statically equivalent nodal point forces. In the work being described here it was considered outside the objectives to try to deduce dynamic loads. It is therefore left to the analyst to select values for the mean acceleration values acting on the plate in its three local axes. These values are entered into the data as ratios of the acceleration assumed to acceleration due to gravity.

New work on the fluid loading of structures is given in Ref.(35) under the heading Fluid - Structure Interaction.

6.5.4 Storage of load values by HYDRO.

6.5.4.1 An examination of Fig. 6.1 shows the assumed form of distribution of the plate element loading into the surrounding stiffener grillage.

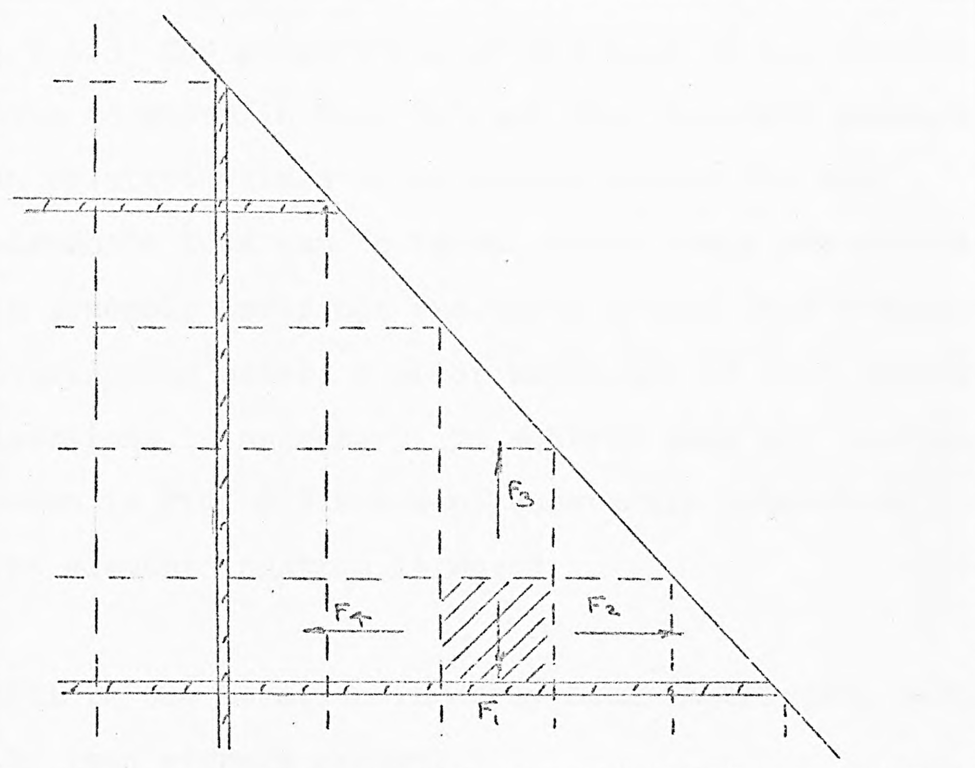


Variable names given to storage locations in array Real used for loads being applied to the current panel.

Fig. 6.3

The value of the load assumed to be taken by any stiffener is inversely proportional to its distance from the element's centre. In those cases where there is no stiffener present to take the allocated out of plane load, the full load is divided amongst the remaining stiffeners. The load on an individual stiffener is divided equally between its ends.

6.5.4.2 In the case of triangular panels the above concept is more difficult to apply for plate elements near the hypoteneuse edge. For a rectangular panel when the distribution of the loading of elements to the right of the last vertical stiffener is being considered, the program can ask the question, "Is there a horizontal stiffener remaining on the unscanned part of the panel which can accept the F3 portion of the load?" This question is answered by checking if the F1 portion is being assigned to the last horizontal stiffener on the plate, in which case there can be no possible stiffener for F3. In the case of a triangular panel as shown is Fig. 6.4 there can exist unscanned horizontal stiffeners which meet the above criteria for rectangular panels but fail to offer a site for the storage of F3, when using the distribution assummed in 6.5.4.1. Elements in the first column of Fig. 6.4 can successfully put their F3 component onto a horizontal stiffener but elements in following columns must redistribute their loads. To accommodate the sensing of whether or not a stiffener is suitably placed, a counter is set up to control the examination of the contents of the index of local freedoms along the next horizontal stiffener to be scanned.



Source of load storing errors for a hydrostatically loaded triangular panel.

Fig. 6.4

If the location of the third order freedom along that stiffener for the next nodal column to be seen on the current row is void then no stiffener can exist to take F3. It is necessary to question the node ahead since the storage for the node vertically above the current node might contain a value and thus give a false answer since it would then be indicating the presence of a stiffener to the left but not necessarily to the right.

A similar loop is necessary to ascertain the availability of a vertical stiffener to the right of the current plate element.

6.5.4.3 The preparation of the load on any element such as shown in Fig. 6.3 and Fig. 6.4 will produce up to eight values to be stored before the next element's load can be found. Since loads are sorted in assembly order but are being formed in a staggered overlapping order, a prior knowledge of load storage locations is necessary. To achieve this six counters shown in Fig. 6.3 are kept constantly updated as the element location is moved.

NEIL is the location in array Real immediately before the load storage segment.

IC is the counter along the lower horizontal stiffener of a pair of stiffeners forming a horizontal bay. This can be simply updated by one for each element seen as the panel is scanned from left to right. Whilst scanning the next horizontal division within the same bay IC continues to return to its original value held at the start of the scan between these two stiffeners. This enables the fresh load values from the new row of elements to be added to the values already stored in consecutive locations from the scan across the previous division of this same bay. On moving into the next higher bay or the first bay after a free edge in the local x direction IC takes on the value of IRV2 + 1, this being the next consecutive location. Loads on the right hand end of stiffener elements are added to the value stored in Real (IC + 1)

NSTORE is the counter along the upper edge stiffener which runs parallel to the stiffener using the counter IC. When all the hydrostatic loads occuring in the bay between these two stiffeners have been stored then what was the upper stiffener of the bay just finished becomes the lower stiffener of the new bay and the variable IC may now take on the value which NSTORE had at the start of the previous bay. As with IC, NSTORE is increased by one with every plate element seen in a scan and reverts to the value it held at the begining of the current bay for the start of each new scan across the plate. To obtain a value for NSTORE at the commencement of each new bay entails finding out the number of storage spaces between the last value of NSTORE and its new initial position. This number must be the number of vertical stiffeners times the number of divisions into which the vertical stiffeners are divided.

ILV and IRV are the counters for storage locations of the lower horizontal component of the load going to the left and right vertical stiffeners respectively. Initially when the bottom left hand plate element between two vertical stiffeners is the current element ILV must equal IC. In this case the plate element load, though divided into its correct number of parts, has two of the parts added into the same location. This will also happen at each corner of the area enclosed by four stiffeners. When scanning the lowest row of plate elements IRV initially takes the value of IC + IXDIV, where IXDIV is the number of divisions in the local x direction between the two vertical stiffeners bounding this segment of the horizontal scan. Subsequently on the lower row of elements in any horizontal block, IRV increases in steps of IXDIV with ILV always taking the last value of IRV. The values of ILV and IRV remain constant between any pair of vertical stiffeners whilst the current element remains on the same row. When scanning rows other than the lowest of a block, because there then is no horizontal stiffener present to upset the storage location count ILV and IRV both increase by one on moving to the next vertical block. Unlike IC and NSTORE when starting a new row counters on vertical stiffeners do not return to a previous value.

ILV2 and IRV2 are counters analogous to ILV and IRV, but where ILV and IRV refer to the lower end of a stiffener element ILV2 and IRV2 refer to the upper end. Initially IRV2 must be IC + ISX, where ISX is the number of nodes along the side of the panel in the x direction. As with ILV so ILV2 takes on the last value of IRV2. Except when scanning the top row of elements, when IRV2 increases in steps of one on moving across a vertical stiffener. Along the top row ILV2 and IRV2 are constant between vertical stiffeners, but IRV2 increases in steps of IXDIV as IRV does when the bottom row is the current row.

The updating of the six counters just described was primarily applicable to rectangular panels. The updating becomes more complicated in the case of triangular panels and some additional programming has had to be included to cope with such conditions as the shaded one in Fig. 6.4.

6.5.4.4 If the structure is subject to point loads as well as hydrostatic loads, the hydrostatic loads are calculated and stored first before reading the point loads which are stored straight after the last hydrostatic load. The variable name for the total number of loads on the panel is NILOAD. In the case of point loads this is included with the data and refers only to those point loads. The subroutine calculates the number of loads being applied as point loads by the hydrostatic head and if point loads are also applied, corrects the value of NILOAD accordingly.

6.6 Loads due to own weight.

6.6.1 As with hydrostatic loadings, the structure may be considered as being subjected to accelerations in the three global directions. If all the accelerations are read in as zero, the vertical acceleration will be automatically corrected to read 1.0, then only gravity will be considered. If the structure's own weight is to be ignored, then the density of the structural material is set to zero. Hydrostatic loads only exist normal to the plate's surface these may be calculated initially then and stored and need not be considered again as anything other than a set of point loadings. Structural weights under the effect of the three accelerations act in the three global directions and each may be resolved into the three local axes, so that those plate element components which act in the local Z direction may be treated and stored like hydrostatic loads. The components in the local x and y directions are evaluated during assembly of the stiffness matrix. The structural load due to weight on one of the degrees of freedom of a plate element is the sum of a quarter of the appropriate resolved weight components for each of the four adjacent plate elements. Assembling a stiffener's contribution to the load vector, will require all three components for each acceleration to be evaluated. Though HYDRO continually refers to the storage locations of the third order freedoms of stiffeners no value of the stiffener is ever put in there, therefore stiffener weight components must be calculated at every stiffener translation degree of freedom. Loads due to own weight do not alter the value of NILOAD as these are too numerous to store individually as point loads and are as easily formed during assembly as an extension to HYDRO.

6.7 Thermal loads.

6.7.1 As explained in 4.3.1, the total strain of the bodies considered by this package consists of two parts, elastic and thermal. Chapter 4 has dealt with the elastic stress - strain relationships and structural elastic stiffness matrices have been developed. In this section we will now consider thermal matrices.

6.7.2 The temperature differences between a gas tanker's hull and its cargo tanks is quite high. Because of the danger of brittle fracture these thermal loads are important. The thermal loads being applied to a cooled or heated structure are derived from the matrix of stresses necessary to supress thermal contraction or expansion. With the type of tank being considered here Becker and Colao Ref. (36) indicate that a conservative estimate of thermal stress for a uniform member is given by

$$\alpha . E . T \qquad \ldots (6.1)$$

For a plate element this can be modified to allow for Poisson's ratio effect to obtain the thermal component of Eq. (4.6) hence

$$\begin{bmatrix} \sigma_{x_T} \\ \sigma_{y_T} \\ \tau_{xy_T} \end{bmatrix} = \frac{E \alpha T}{(1-\nu)} \begin{bmatrix} -1 \\ -1 \\ 0 \end{bmatrix} \qquad \ldots (6.2)$$

which is of the form

$$\sigma = \alpha T \left[D_T \right] \qquad \ldots(6.3)$$

Becker and Colao explain that Eq. (6.1) is not suitable for estimating stresses in membrane tanks. This is because the construction of a membrane tank is such that stiffening of the skin is obtained by putting folds into the material to form channels. Contraction of the flat rectangular panel enclosed by a set of four channels will induce into the tips of the channels a large stress due to the bending of the walls of the channels.

6.7.3 Thermal stiffness matrix.

From the unit-displacement theorem

$$\underline{S} = \int e^T \sigma \, dv$$

Substituting Eq. (6.3) gives

$$S = \alpha T \int e^T \left[D_T \right] dv \qquad \ldots(6.4)$$

From the reasoning of Chapter 4 relating to Eq. (4.7) for unit displacement we can now substitute b^T for e^T and Eq. (6.4) will give Q, the vector of loads to suppress thermal distortion which may be written

$$Q = \alpha T h \qquad \ldots(6.5)$$

where h is the thermal matrix and

$$h = \int b^T D_T \, dv \qquad \ldots(6.6)$$

6.7.4 Consider a rectangular plate element. From the formulation used previously for a stiffness matrix obtained by using distortion patterns we have

$$e = \left[B \right] \left[A^{-1} \right] u \qquad \ldots(4.40)$$

and

$$e = \left[b \right] u \qquad \ldots(4.7)$$

Substituting Eq.(4.7) into Eq.(4.40) we obtain

$$[b] = [B][A^{-1}] \qquad \ldots(6.7)$$

Performing the matrix multiplication $[B].[A^{-1}]$ where $[B]$ is defined in Eq.(4.34) we obtain Eq.(6.8).

Post multiplying $[[B][A^{-1}]]^T$ by $[D_T]$ we obtain Eq.(6.9).

$$\frac{E}{(1-\nu)}\begin{bmatrix} x(\nu-1)/4ab + 1/4b \\ y(\nu-1)/4ab + 1/4a \\ x(1-\nu)/4ab + 1/4b \\ y(1-\nu)/4ab - 1/4a \\ x(1-\nu)/4ab - 1/4b \\ y(1-\nu)/4ab + 1/4a \\ x(\nu-1)/4ab - 1/4b \\ y(\nu-1)/4ab - 1/4a \end{bmatrix} \qquad \ldots(6.9)$$

Integrating Eq.(6.9) with respect to x and y for a constant thickness we get Eq.(6.10).

$$\frac{Et}{(1-\nu)}\begin{bmatrix} x^2y(\nu-1)/8ab + xy/4b \\ xy^2(\nu-1)/8ab + xy/4a \\ x^2y(1-\nu)/8ab + xy/4b \\ xy^2(1-\nu)/8ab - xy/4a \\ x^2y(1-\nu)/8ab - xy/4b \\ xy^2(1-\nu)/8ab + xy/4a \\ x^2y(\nu-1)/8ab - xy/4b \\ xy^2(\nu-1)/8ab - xy/4a \end{bmatrix} \qquad \ldots(6.10)$$

$\frac{x}{4ab} - \frac{1}{4b}$	$-\frac{\nu y}{4ab}$	$-\frac{x}{4ab} - \frac{1}{4b}$	$\frac{\nu y}{4ab}$	$-\frac{x}{4ab} + \frac{1}{4b}$	$\frac{\nu y}{4ab}$	$\frac{x}{4ab} + \frac{1}{4b}$	$-\frac{\nu y}{4ab}$
$-\frac{\nu x}{4ab}$	$\frac{y}{4ab} - \frac{1}{4a}$	$\frac{\nu x}{4ab}$	$\frac{-y}{4ab} + \frac{1}{4a}$	$\frac{\nu x}{4ab}$	$-\frac{y}{4ab} - \frac{1}{4a}$	$-\frac{\nu x}{4ab}$	$\frac{y}{4ab} + \frac{1}{4a}$
$-\frac{1}{4a}$	$-\frac{1}{4b}$	$\frac{1}{4a}$	$-\frac{1}{4b}$	$-\frac{1}{4a}$	$\frac{1}{4b}$	$\frac{1}{4a}$	$\frac{1}{4b}$

Matrix b relating strains to displacements for a distortion pattern element. ...(6.8)

Substituting the limits of integration a to -a for x and b to -b for y, we obtain the vector (6.11) being the value of $\int b^T D_T \, dv$ of Eq.(6.5), the thermal stiffness matrix.

$$\frac{E t}{(1-\nu)} \begin{bmatrix} a \\ b \\ a \\ -b \\ -a \\ b \\ -a \\ -b \end{bmatrix}$$

...(6.11)

Eq.(6.5)may now be rewritten

$$Q = \frac{E . t . \alpha . T . a}{(1 - \nu)} \begin{bmatrix} 1 \\ b/a \\ 1 \\ -b/a \\ -1 \\ b/a \\ -1 \\ -b/a \end{bmatrix} \quad \text{...(6.12)}$$

Where Q is the vector of thermal loads of a rectangular element derived using distortion shape functions. The order of the vector coefficients is as in Fig. 4.5.

The thermal stiffness matrix and thermal load vector may also be derived using the notation of 4.3 as follows. Substituting Eq.(4.21) for unit displacement into Eq.(6.4) gives Eq.(6.13).

$$Q = \alpha T \int [B]^T [D_T] \, dv \qquad \ldots(6.13)$$

Post multiplying $[B]^T$ as defined in Eq.(4.21) by D_T obtain the vector (6.14).

$$\frac{E\alpha T}{(1-\nu).4ab} \begin{bmatrix} (a-x) \\ (b-y) \\ (a+x) \\ -(b-y) \\ -(a-x) \\ (b+y) \\ -(a+x) \\ -(b+y) \end{bmatrix} \qquad \ldots(6.14)$$

Integrating (6.14) we obtain (6.15) and substituting the limits a to −a for x and b to −b for y we again obtain the thermal load vector Q.

$$\frac{\alpha T}{4ab} \begin{bmatrix} axy - x^2y/2 \\ bxy - xy^2/2 \\ axy + x^2y/2 \\ -bxy + xy^2/2 \\ -axy + x^2y/2 \\ bxy + xy^2/2 \\ -axy - x^2y/2 \\ -bxy - xy^2/2 \end{bmatrix} \ldots(6.15) \qquad Q = \frac{E t \alpha T a}{(1-\nu)} \begin{bmatrix} 1 \\ b/a \\ 1 \\ -b/a \\ -1 \\ b/a \\ -1 \\ -b/a \end{bmatrix} \ldots(6.16)$$

6.7.5 Consider a right angled triangular plate element, with the degrees of freedom order as in Fig. 4.17. Substituting $[B]$ as defined in Eq.(4.59) and $[D_T]$ as defined by Eq.(6.3) into Eq.(6.13), we obtain

$$Q = \frac{\alpha T E}{2A(1-\nu)} \int \begin{bmatrix} x_{32} & 0 & -y_{32} \\ 0 & -y_{32} & x_{32} \\ 0 & 0 & y_{31} \\ 0 & y_{31} & 0 \\ x_{21} & 0 & 0 \\ 0 & 0 & x_{21} \end{bmatrix} \begin{bmatrix} -1 \\ -1 \\ 0 \end{bmatrix} dv \qquad \ldots(6.17)$$

Completing the matrix multiplication of (6.17) gives

$$Q = \frac{\alpha T E}{2A(1-\nu)} \int \begin{bmatrix} -x_{32} \\ y_{32} \\ 0 \\ -y_{31} \\ -x_{21} \\ 0 \end{bmatrix} dv \qquad \ldots(6.18)$$

All the terms of the vector are independent of any function of x or y, therefore the integration reduces to At = volume of the triangular element.

$$\therefore \quad Q = \frac{\alpha T E t}{2(1-\nu)} \begin{bmatrix} -x_{32} \\ y_{32} \\ 0 \\ -y_{31} \\ -x_{21} \\ 0 \end{bmatrix} \qquad \ldots(6.19)$$

6.7.6 Consider a beam element.

6.7.6.1 As explained previously the beam elements used in this package are considered to be eccentrically loaded through the node of the adjoining plate element.

The thermal stiffness matrix of an eccentric beam can be easily obtained by considering the value of the forces and moments necessary to suppress thermal distortion. It should be noted that in this structure each panel including stiffeners is considered to be at a constant temperature throughout, though temperature differences may exist between panels.

In this work a beam is considered subject to only thermal stresses in the longitudinal direction and rotations normal to the plate's surface contained in the longitudinal plane of the stiffener.

6.7.6.2 To suppress a beam's change in length due to a temperature variation there must be applied at one end of the beam a force of $E.\alpha.T.A$ and a force of equal value acting in the opposite direction at the beam's other end.

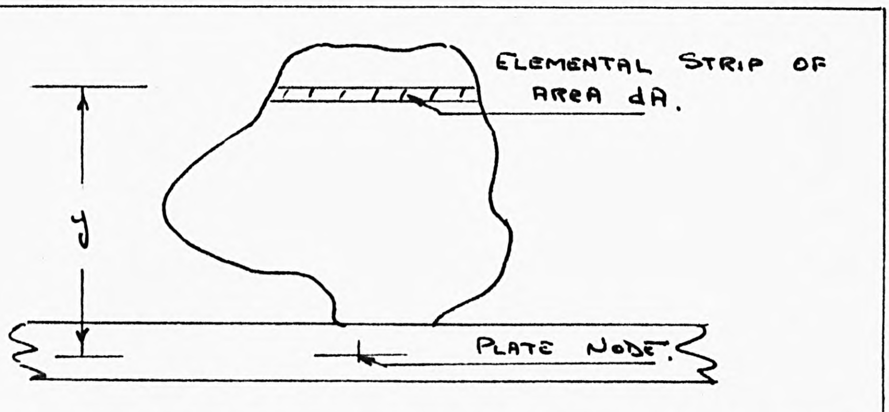


Fig. 6.5

In suppressing this change of length the force is considered to act at the adjoining plate's node.

This asymmetric loading will cause a moment which must be included in the thermal stiffness matrix. Consider a cross section of a uniform beam to be shown as in Fig. 6.5.

The force on the strip = $E \alpha T \, dA$

The moment of this force about the plate's node =

$$E \alpha T \, dA \cdot y$$

Integrating w.r.t. dA $\cdot$ $E \alpha T A . E_c$

Therefore, thermal moment necessary to suppress asymmetric thermal loading =

$$-E \alpha T A E_c \qquad \ldots(6.20)$$

The beam's thermal stiffness matrix may be written as

$$E \alpha T A \begin{bmatrix} 1 \\ 0 \\ -E_a \\ -1 \\ 0 \\ E_c \end{bmatrix} \qquad \ldots(6.21)$$

where E_c is the eccentricity of the beam about the plate's node.

6.7.7 Evaluation of the thermal load vector.

We have seen in 4.3 that in elastic systems there is a relationship between P the vector of loads applied and U the vector of resulting displacements such that P = KU, where K is the elastic structural stiffness matrix. If Q is the vector of thermal loads to be applied to the structure to suppress thermal distortion, then the vector P may be modified to take account of thermal loading by rewriting the relationship between applied loads and displacements as

$$P - Q = [K] U \qquad \ldots(6.22)$$

As was seen in chapter 5, forming the load vector is progressed at the same time as assembling the stiffness matrix, such that the one load coefficient corresponding to the current row of the matrix is the only input made to the load vector until the next row. The thermal load vector of an element does not lend itself to assembly in this manner and a modification must be made to the concept of a thermal load vector for an element changing it to a thermal load vector for a degree of freedom.

No attempt is made to form and store the thermal loads at the start as with hydrostatic loads. The plate thermal load values are dependent upon material thicknesses and element length and breadth. The current values of all these items are available in PLATEK. This subroutine is therefore used to obtain the thermal load coefficients as it is forming the row of the stiffness matrix.

6.7.7.1 Thermal load vector for a group of rectangular elements.

Consider a plate node A which is not on the perimeter of the plate, having degrees of freedom 1 and 2 with the dimensions and thicknesses as shown in Fig. 6.6.

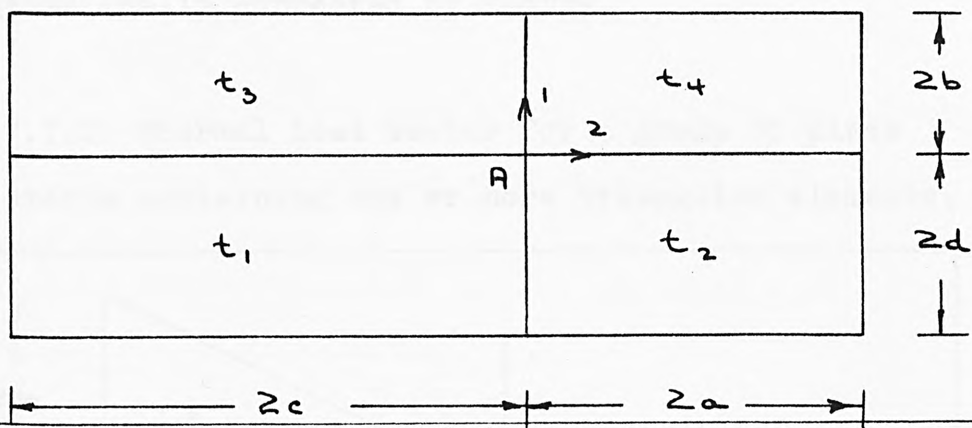


Fig. 6.6

Then using Eq.(6.11) and the order of degrees of freedom shown in Fig. 4.5, the load at freedom 1 is

$$-E t_1 \alpha T c / (1-\nu)$$

contributed by the bottom left hand element of Fig. 6.6,

$$-E t_2 \alpha T a / (1-\nu)$$

contributed by the bottom right hand element of Fig. 6.6,

$$E t_3 \alpha T c / (1-\nu)$$

contributed by the top left hand element of Fig. 6.6,

$$E t_4 \alpha T a / (1-\nu)$$

contributed by the top right hand element of Fig. 6.6.

Summing these components we obtain the total thermal load at freedom 1 due to plate elements of Fig. 6.6 as

$$\frac{E \alpha T}{(1-\nu)} \left(-t_1 c - t_2 a + t_3 c + t_4 a\right) \quad \ldots(6.23)$$

Similarly the thermal load at freedom 2 is

$$\frac{E \alpha T}{(1-\nu)} \left(-t_1 d + t_2 d - t_3 b + t_4 b\right) \quad \ldots(6.24)$$

In cases where node A is on the perimeter of the panel so that one or more of the dimensions are non existant, then Eq.(6.23) and Eq.(6.24) are automatically corrected by PLATEK.

6.7.7.2 Thermal load vector for a group of plate elements containing one or more triangular elements.

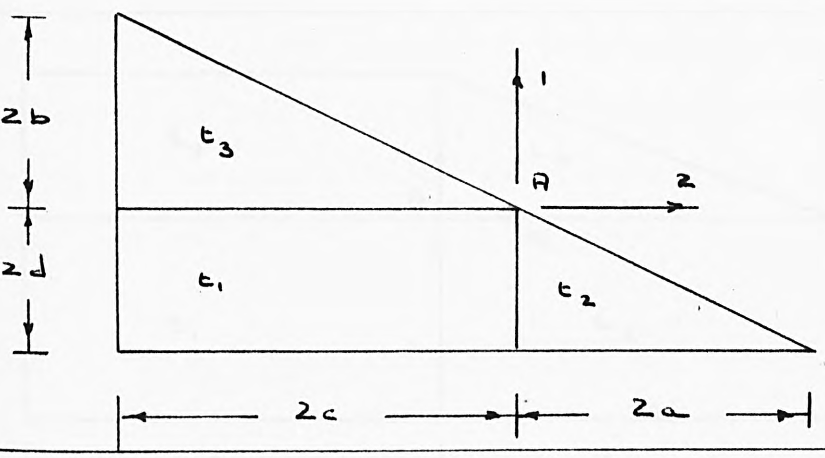


Fig. 6.7

Consider the case where node A is on the perimeter of a triangular panel having the dimensions and thicknesses as shown in Fig. 6.7. The thermal load at freedom 1 can be formed as follows;

From Eq.(6.12)

$$-E t_1 \alpha T c \;/\; (1-\nu)$$

contributed by the only rectangular element.

From Eq.(6.19)

$$-E t_2 \alpha T a \;/ (2\,(1-\nu))$$

contributed by the bottom right hand triangular element.

The contribution of the top triangular element is from its 3rd freedom, from Eq.(6.19) we see that this is zero. Therefore the thermal load at freedom 1 of node A in Fig. 6.7 is

$$\frac{E\alpha T}{(1-\nu)}\left(-t_1 c - t_2 a/2\right) \qquad \ldots(6.25)$$

Similarly at freedom 2 the load is

$$\frac{E\alpha T}{(1-\nu)}\left(-t_1 d - t_3 b/2\right) \qquad \ldots(6.26)$$

Consider the case where node A is one element away from the hypoteneuse of a triangular panel as shown in Fig. 6.8.

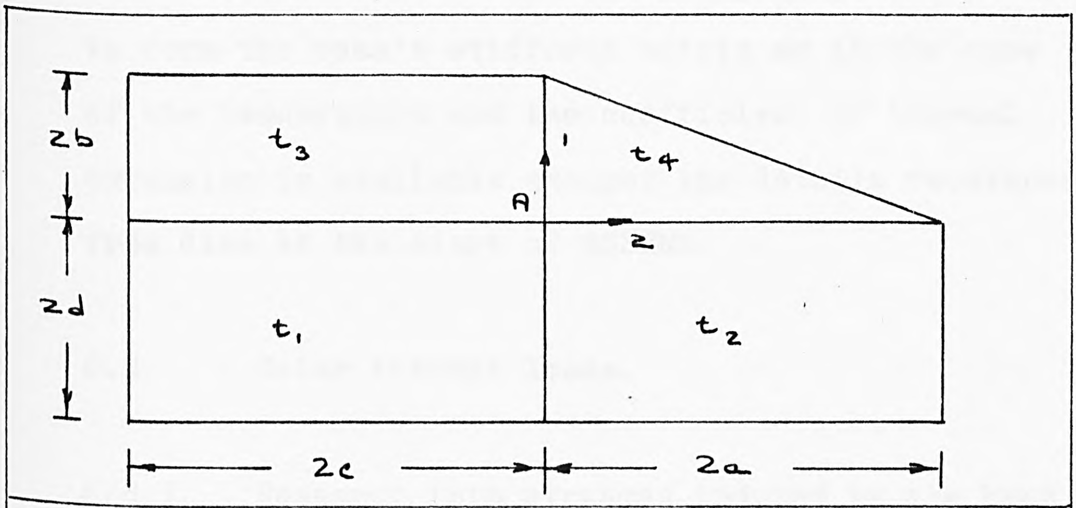


Fig. 6.8

Using Eq.(6.12) the thermal load at freedom 1 on Fig. 6.8 can be formed by summing

$-Et_1\alpha Tc/(1-\nu)$ contributed by the bottom left hand rectangular element,

$-Et_2\alpha Ta/(1-\nu)$ contributed by the bottom right hand rectangular element,

$Et_3\alpha Tc/(1-\nu)$ contributed by the topleft hand rectangular element, and

$Et_4\alpha Ta/(2(1-\nu))$ contributed by the only triangular element as given by Eq.(6.19).

Thus the thermal load at freedom 1 of Fig. 6.8 is $\frac{E\alpha T}{(1-\nu)}\left(-t_1 a - t_2 a + t_3 c + t_4 a/2\right)$

Similarly for freedom 2 the thermal load is

$$\frac{E\alpha T}{(1-\nu)}\left(-t_1 d + t_2 d - t_3 b + t_4 b/2\right)$$

6.7.7.3 Thermal load vector for a beam element.

Like the plate elements, the beam element thermal load is formed in association with the stiffness matrix. This though is not in PLATEK but directly in ASSEMB. All the information to form a beam's thermal loading is already available ready to form the beam's stiffness matrix or in the case of the temperature and the coefficient of thermal expansion is available amongst the details recovered from disc at the start of ASSEMB.

6.8 Solar thermal loads.

6.8.1 Research into stresses induced by the heat of the sun in ship structures has been carried out by Meriam et al Ref.(37). This shows that stresses of 10,000 lbf/in^2 are well within operational conditions. This package allows any panel to have its own unique mean temperature difference. So that plating exposed to direct sunlight may be given one temperature range whilst different values may be applied to the shell in contact with the sea and the tank in contact with the cargo.

6.9 Total restraints.

6.9.1 Panels are assembled as stiffened two dimensional structures and only when being assembled into the three dimensional structure are any third order translation freedoms added other than those genuine freedoms due to stiffeners. This means that there are no artificial restraints to be applied.

For ease of data preparation restraints due to structural and symmetrical restrictions are defined by two digits per restraint. The first being the node's local number and the second the order of freedom being restrained at that node. These may be internal as well as on plate edges. Though for convenience the position of each restraint is put into the data as just described, the program requires the restraints to be defined by the numerical position in the assembly order. This is achieved by making the index which lists all the degrees of freedom on the panel, store all the reference numbers for each node in blocks of equal length irrespective of the number of freedoms at any node. The node's local number is then the number of blocks into the index which will contain the assembly order number.

6.10 Partial restraints.

6.10.1 Partial restraint of degrees of freedom may be obtained by the use of springs. Loads other than restraints may be applied to the structure through springs. Springs in this package have a special role during the inversion stage and discussion of their action will be gone into more fully in chapter 9.

7. Reduction in size of matrix whilst still assembling.

7.1 Best known existing techniques.

7.1.1 Perhaps the two best known methods for reducing the stiffness matrix size are the wavefront or frontal solution technique and the substructuring method.

7.1.2 The wavefront technique is described in a paper "A Frontal Solution Program for Finite Element Analysis" by Irons (38). Though this technique effectively reduces the size of the stiffness matrix, this was not the primary object and this leads to two disadvantages:

a. The process of full matrix assembly has to be continued until a particular degree of freedom is itself fully assembled, which can lead to a large amount of core space being occupied by partially assembled degrees of freedom, which is not acceptable if there is insufficient space available.

b. Because of the random nature of the removal of inactive degrees of freedom it is necessary to maintain an extensive check on the history of the freedoms in what is called a housekeeping routine.

7.1.2.1 This requirement to maintain an extensive housekeeping routine can be dispensed with as in Sabir's work (39) where no attempt is made to remove an inactive row of the stiffness matrix before the current triangle of coefficients passes clear of that line.

7.1.3 Of the other method, Prezemienieki (32) describes an approach for reducing the overall stiffness matrix by dividing the structure into substructures, forming fully the stiffness matrix of each substructure and then re-expressing each substructure matrix in terms of its boundary degrees of freedom. The substructures are then treated as complex elements to assemble the whole. It can be seen from examples given in his book that the part of a structure suitable for consideration as a substructure is one which has a small ratio of bandwidth to total number of degrees of freedom. These are generally structures which may appear as appendages to the remainder of the whole such as a wing to the main body of an aircraft or a part of the pressure hull to the remainder of a submarine's cylindrical hull.

7.1.4 The full stiffness matrix of each substructure must obviously be not larger than the remaining core space. If it should be larger then further structural partitioning must be undertaken until the space requirements are complied with. Although these words seem to apply to the proposed method, the difference is that the proposed method requires only core space for the stiffness matrix in terms of the boundary degrees of freedom and not the full matrix. This device gains in advantage as the substructures grow in size, therefore less substructuring is necessary.

7.2 Advantage of the stiffness vector.

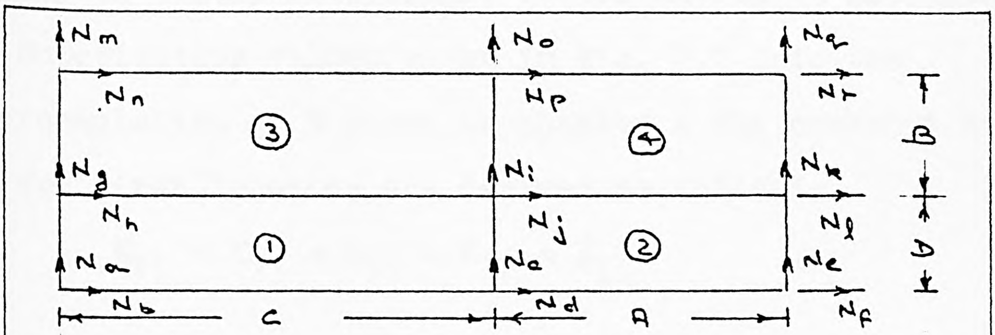
7.2.1 Conventional stiffness matrix assembly methods use a process of overlapping element stiffnesses, whereby though each element's contribution is fully assembled before going onto the next, individual degrees of freedom which are shared by more than one element may not be completed until much later. As mentioned before, in this work a concept is introduced of a stiffness vector which allows each row of the final stiffness matrix to be assembled in its entirety. This allows a significant reduction in housekeeping routines and a smaller stiffness matrix will be formed at each stage when the removal of inactive degrees of freedom is due. In this case a complete element may not be fully assembled until much later. This is of no consequence as all the subsequent calculations and results are related to the reference number automatically allocated to each degree of freedom.

7.3 Types of vectors.

7.3.1 To accommodate this concept and express it in normal terms, three types of element stiffness vectors have been developed from the conventional beam and plate elements, derived in chapter 4. One is for a beam having three nodes (two of which may coincide) with each node having 3 degrees of freedom. The other two are plate elements. One is a vector for a degree of freedom of a composite rectangular element formed from up to 4 suitably adjacent rectangular elements into which the structure is divided. This element has either 2 or 3 nodes per side and in most cases an additional node within the element. Each of the nodes having only 2 inplane freedoms. The other is for use on or in the vicinity of the hypoteneuse of a triangular panel and has two forms depending on whether it is a composite of one triangular and three rectangular elements or two triangular and one rectangular.

7.4 Derivation of stiffness vector for a rectangular panel.

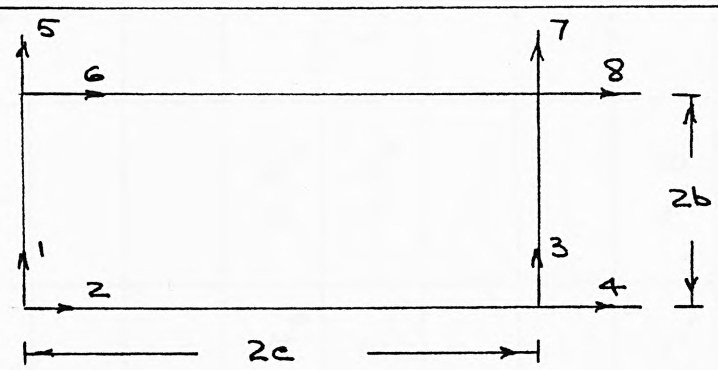
7.4.1 Consider four plate elements enclosing a current node. Let the reference numbers of all the plate elements and degrees of freedom be as shown in Fig. 7.1.



Numbering of freedoms round a node.

Fig. 7.1

If the stiffness matrix of a basic plate element having 8 degrees of freedom arranged as Fig. 7.2 is denoted by K, then in the assembled stiffnes matrix



Order of numbering degrees of freedom.

Fig. 7.2

of the structure containing the four elements shown in Fig. 7.1, the leading diagonal term of the N_i^{th} row will be

$$\Sigma \quad K(7,7)_1 + K(5,5)_2 + K(3,3)_3 + K(1,1)_1 \quad \ldots(7.1)$$

where the suffix refers to the element making that contribution. Similarly the last term of the N_j^{th} row is just $K(2,8)_4$. Fig. 7.3 shows the full contribution to the N_i^{th} and N_j^{th} rows.

7.4.1.1 We can see therefore that all this information may be stored in a single vector having 19 locations. Substituting values given in Fig. 7.1 into the formulation of K given in chapter 4 the contents of the first location are derived as follows,

$$K_{11} = K_{33} = K_{55} = K_{77} = K_{ii}$$

$$= Et\left(\frac{a}{12\,b} + \frac{a}{4b(1-\nu^2)} + \frac{b}{8\,a\,(1+\nu)}\right)\ldots(7.2)$$

	Column Number.									
	i	j	k	l	m	n	o	p	q	r
Row i	$7,7_1$	$7,8_1$								
	$5,5_2$	$5,6_2$	$5,7_2$	$5,8_2$						
	$3,3_3$	$3,4_3$			$3,5_3$	$3,6_3$	$3,7_3$	$3,8_3$		
	$1,1_4$	$1,2_4$	$1,3_4$	$1,4_4$			$1,5_4$	$1,6_4$	$1,7_4$	$1,8_4$
Row j		$8,8_1$								
		$6,6_2$	$6,7_2$	$6,8_2$						
		$4,4_3$			$4,5_3$	$4,6_3$	$4,7_3$	$4,8_3$		
		$2,2_4$	$2,3_4$	$2,4_4$			$2,5_4$	$2,6_4$	$2,7_4$	$2,8_4$

Coefficient Numbers of Stiffness Matrix Making up Stiffness Vector.

Fig. 7.3

By inspection we can see the values for a and b are as shown in Fig. 7.4

i	a	b
1	C/2	D/2
2	A/2	D/2
3	C/2	B/2
4	A/2	B/2

Fig. 7.4

Therefore the first location value is

$$Et*\Big\{\frac{C}{2D}+\frac{C}{4D(1-\nu^2)}+\frac{D}{8C(1+\nu)}$$
$$+\frac{A}{12D}+\frac{A}{4D(1-\nu^2)}+\frac{D}{8A(1+\nu)}$$
$$+\frac{C}{12B}+\frac{C}{4B(1-\nu^2)}+\frac{B}{8C(1+\nu)}$$
$$+\frac{A}{12B}+\frac{A}{4B(1-\nu^2)}+\frac{B}{8A(1+\nu)}\Big\}$$

$$= Et*\Big\{(C+A)(1/D+1/B)(1/12+1/(4(1-\nu^2)))$$
$$+(D+B)(1/C+1/A)/(8(1+\nu))\Big\}$$

...(7.3)

The reciprocals may be removed by substituting

$$1/A = X$$
$$1/B = Z$$
$$1/C = W$$
$$1/D = Y$$

If the current node is on an edge at least one of A,B,C and D will be zero. In such cases the corresponding X,Z,W or Y is put to zero.

Making further substitutions of

$$U = Et/((1+\nu)8)$$

$$F = (1/(4(1-\nu^2))+1/12)8(1+\nu)$$

$$H = W + X$$

$$P = D + B$$

$$Q = Y + Z$$

and $$G = C + A.$$

7.4.2 Then the above value for the first location reduces to

PHU + GFQU

Similarly the contents of the other odd numbered locations of the vector are in order:

-AFQU - XPU

CZFU - BWU

BHU - ZFGU

AZFU - BXU

GQU + PHFU

AQU - FXPU

BWFU - CZU

-BFHU - ZGU

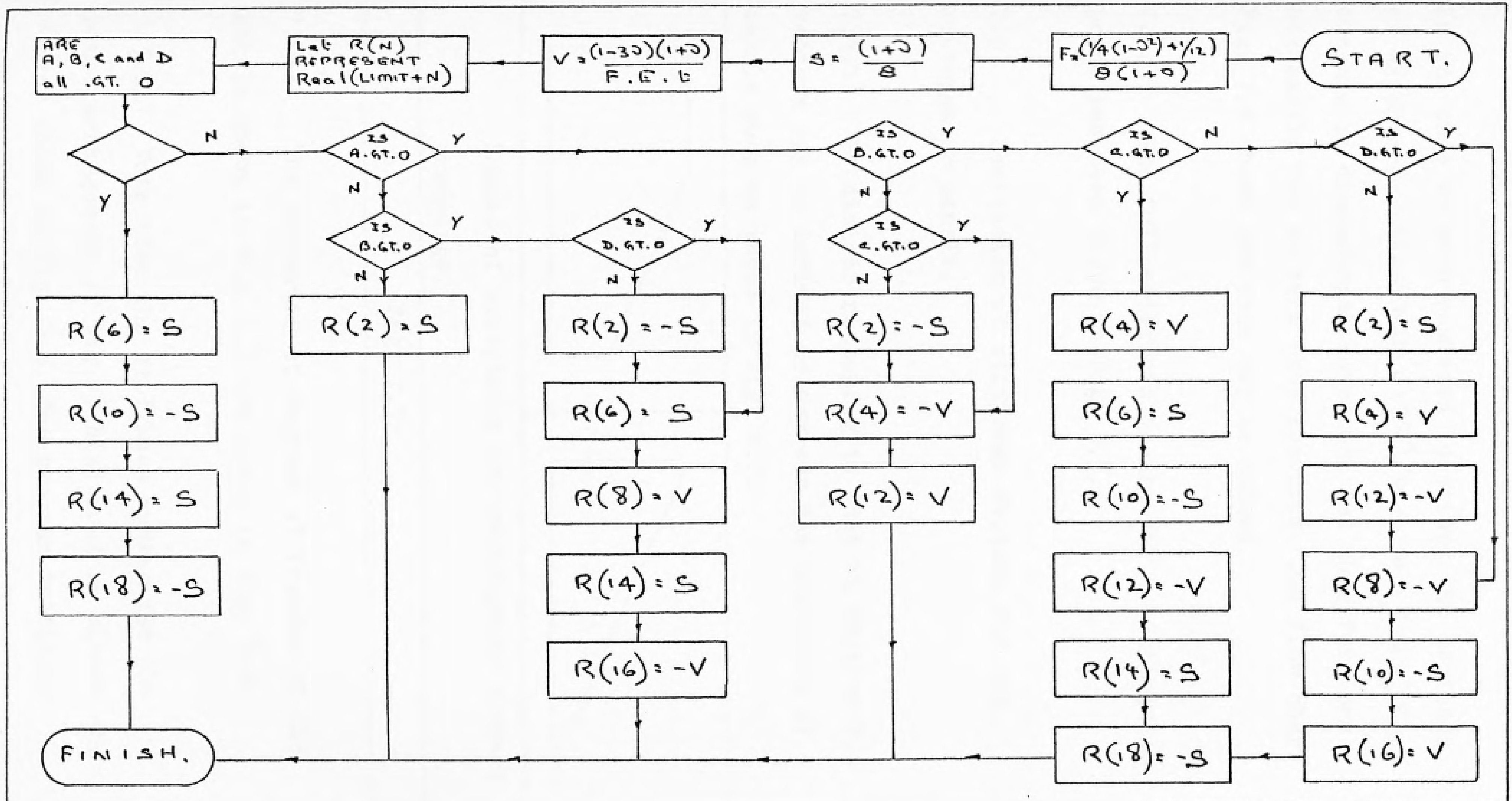
BFXU - AZU

The contents of even numbered locations are shown in Fig. 7.5. In the case of odd numbered locations the contents can be made to calculate themselves to zero when necessary for an edge node.

Location Number	Edge Status.								
A	> 0	> 0	= 0	= 0	= 0	> 0	> 0	> 0	> 0
B	> 0	> 0	> 0	> 0	= 0	= 0	= 0	> 0	> 0
C	> 0	> 0	> 0	> 0	> 0	> 0	= 0	= 0	= 0
D	> 0	= 0	> 0	= 0	> 0	> 0	> 0	> 0	= 0
2	0	0	0	-S	S	0	-S	0	S
4	0	V	0	0	0	-V	-V	0	V
6	S	S	S	S	0	0	0	0	0
8	0	0	V	V	0	0	0	-V	-V
10	-S	-S	0	0	0	0	0	-S	-S
12	0	-V	0	0	0	V	V	0	-V
14	S	S	S	S	0	0	0	0	0
16	0	0	-V	-V	0	0	0	V	V
18	-S	-S	0	0	0	0	0	-S	-S

Contents of Stiffness Vector in Even Numbered Locations.

Fig. 7.5



Derivation of stiffness vector's even numbered locations.

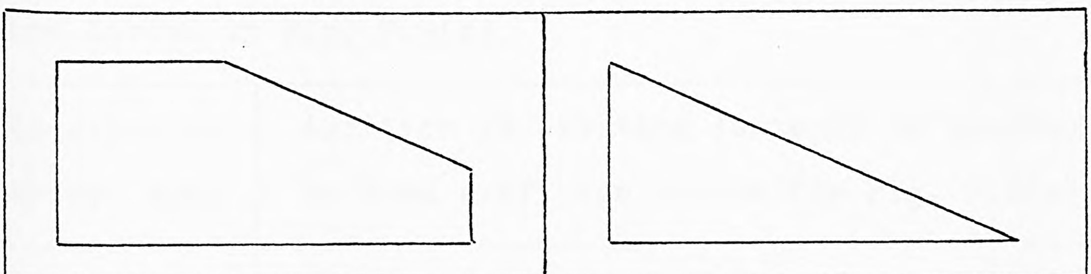
Fig. 7.6

In the case of even numbered locations the contents of which are either $FU(1-3\nu)/8$ or $FU(1+\nu)/8$ contain no dimensional component and therefore are not easily put to zero when required. The flow chart Fig. 7.6 shows how this may be solved.

7.4.3 The full stiffness vector for a rectangular panel has been shown in Fig. 5.8.

7.5 Derivation of stiffness vectors for non rectangular panels.

7.5.1 By similar reasoning to that in section 7.4 vectors can be derived to express the stiffness of panels such as those in Fig. 7.7.



Shapes of acceptable non rectangular element groupings.

Fig. 7.7

7.5.2 The numbering of degrees of freedom of the panels shown in Fig. 7.7 are given in Fig. 7.8.

7.5.2.1 Note that the stiffness vector for the panel arrangement of Fig. 7.8(a) must be almost the vector shown in Fig. 5.8. The program initially processes the four elements as if they were all rectangles and forms the appropriate stiffness vector for such a group.

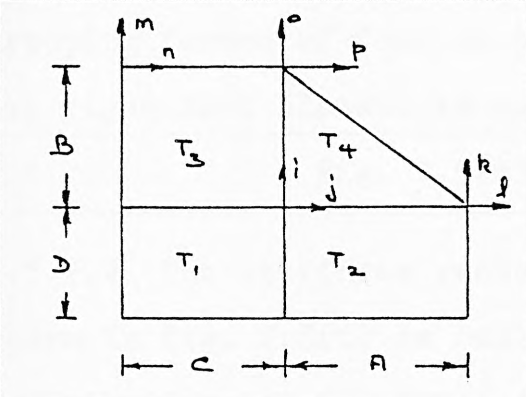


Fig. 7.8(a)

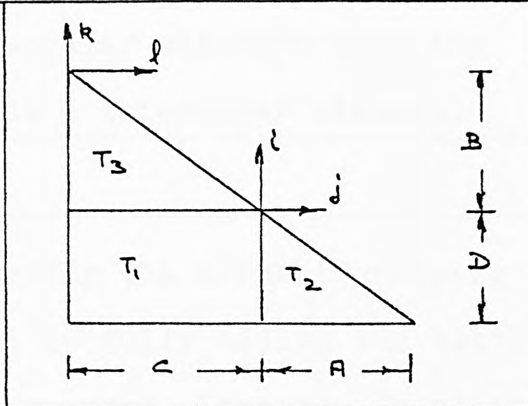


Fig. 7.8(b)

That vector is then modified by the removal of the contribution made by the fourth rectangular element and the addition of the vector shown in chapter 4 due to a single triangular element. These modifications are listed in Fig. 7.9(a).

Location in array Real	Addition to existing contents of location to form stiffness vector for Fig. 7.8(a).
LIMIT + 1	$E.T_4.A/(2(1-\nu^2).B) + E.T_4.B/(4(1+\nu).A)$
LIMIT + 2	$\nu.E.T_4/(2(1-\nu^2)) + E.T_4/(4(1+\nu))$
LIMIT + 3	$-E.T_4 B/(4(1+\nu).A)$
LIMIT + 4	$-\nu.E.T_4/(2(1-\nu^2))$
LIMIT + 7	$-E.T_4.A/(2(1-\nu^2).B)$
LIMIT + 8	$-E.T_4/(4(1+\nu))$
LIMIT + 11	$E.T_4(B/(A(1-\nu)) + A/2.B)/(2(1+\nu))$
LIMIT + 12	$-E.T_4/(4(1+\nu))$
LIMIT + 13	$-E.T_4 B/(2(1-\nu^2).A)$
LIMIT + 16	$-\nu.E.T_4/(2(1-\nu^2))$
LIMIT + 17	$-E.T_4.A/(4(1+\nu).B)$

The contents of locations LIMIT + 9, + 10, + 18 and + 19 are all put to zero as they no longer exist for this non rectangular shape.

Modifications to the stiffness vector of an element grouping formed of four rectangular elements when the top right hand element is made a triangular element.

Fig. 7.9(a)

7.5.2.2 The stiffness vector for the element grouping shown in Fig. 7.8(b) is built up fully adding the vector contributions of the three component elements. No attempt is made in this case to load another configuration and then modify it.

Location in array Real	Contents of location to form the stiffness vector for Fig. 7.8(b).
LIMIT + 1	$E.T_2.A/(2(1-\nu^2).D) + E.T_3\,B/(4(1+\nu).C)$ $+ E.T_1\left(\frac{C}{D}\left(1/4(1-\nu^2) + 1/12\right) + D/(8.C.(1+\nu))\right)$
LIMIT + 2	$E.T_1\left(1/(8(1+\nu)) + \nu/(4(1-\nu^2))\right)$
LIMIT + 4	$E.T_3/(4(1+\nu))$
LIMIT + 11	$E.T_1\left(\frac{D}{C}\left(1/4(1-\nu^2) + 1/12\right) + C/(8.D.(1+\nu))\right)$ $+ E.T_2.A/(4.(1+\nu).D) + E.T_3.B/(2(1-\nu^2)C)$
LIMIT + 12	$E.T_3\nu/(2(1-\nu^2))$

The formulation of terms in the stiffness vector of an element grouping of two triangles and one rectangle.

Fig. 7.9(b)

7.6 Adjusting the element vector length.

7.6.1 The variable KN is the location of the number of coefficients that a particular vector will contain. Because of the plate edge effects this will not be a constant. In cases where voids appear in the vector before the last coefficient, all coefficients to the right of a void are moved to the left until all voids are full. The routine will discriminate between zeros and voids.

7.6.2 KN is set initially to a maximum value of 10. This applies to a plate's first degree of freedom at a node away from the panel edge. On the edges of a rectangular panel the first freedom has values of 8 along the left hand side, 6 on the right hand side, 4 on the top edge, and two at the top right hand corner. In the case of a triangular panel the same values are assigned to KN for nodes other than those on or next to the hypoteneuse. In their cases they have respectively the values 4 and 8 unless the node is at the top of the left hand side when KN becomes 2 or the node immediately below that, when it has the value 6. KN for the second degree of freedom at a node is obtained by reducing the existing KN by 1 at the start of assembling the plate's second freedom coefficients.

7.7 Some difficulties.

7.7.1 The papers of both Irons (38) and Melosh (40) deal only with the solution of already known stiffness matrices. Section 8.9 of Hinton and Owen (41) gives a program for the frontal solution and in the preamble states "It is assumed that the element stiffness matrices and load vector have been generated elsewhere."

7.7.2 Chapter 11 of Finite Element Progamming (41) contains at section 11.6, "A condition which is more difficult to accommodate in the frontal solution is the situation where a different number of degrees of freedom exist at each node, extensive programming is necessary". Which is not very helpful when dealing with practical structures containing stiffeners.

7.7.3 At first sight there seems to be no literature dealing with carrying the wave across a boundary and onto an adjacent inclined panel. On further consideration it becomes clear that the problem does not occur. Two reasons can be seen for this; a) the membrane plate elements are usually written with dummy third translation degrees of freedom which are then suppressed as necessary and b) crossing a boundary becomes an assembly problem which the wave front can take in its stride by using the time wasting procedure of going through the whole problem once, before assembly.

7.8 The reduction process.

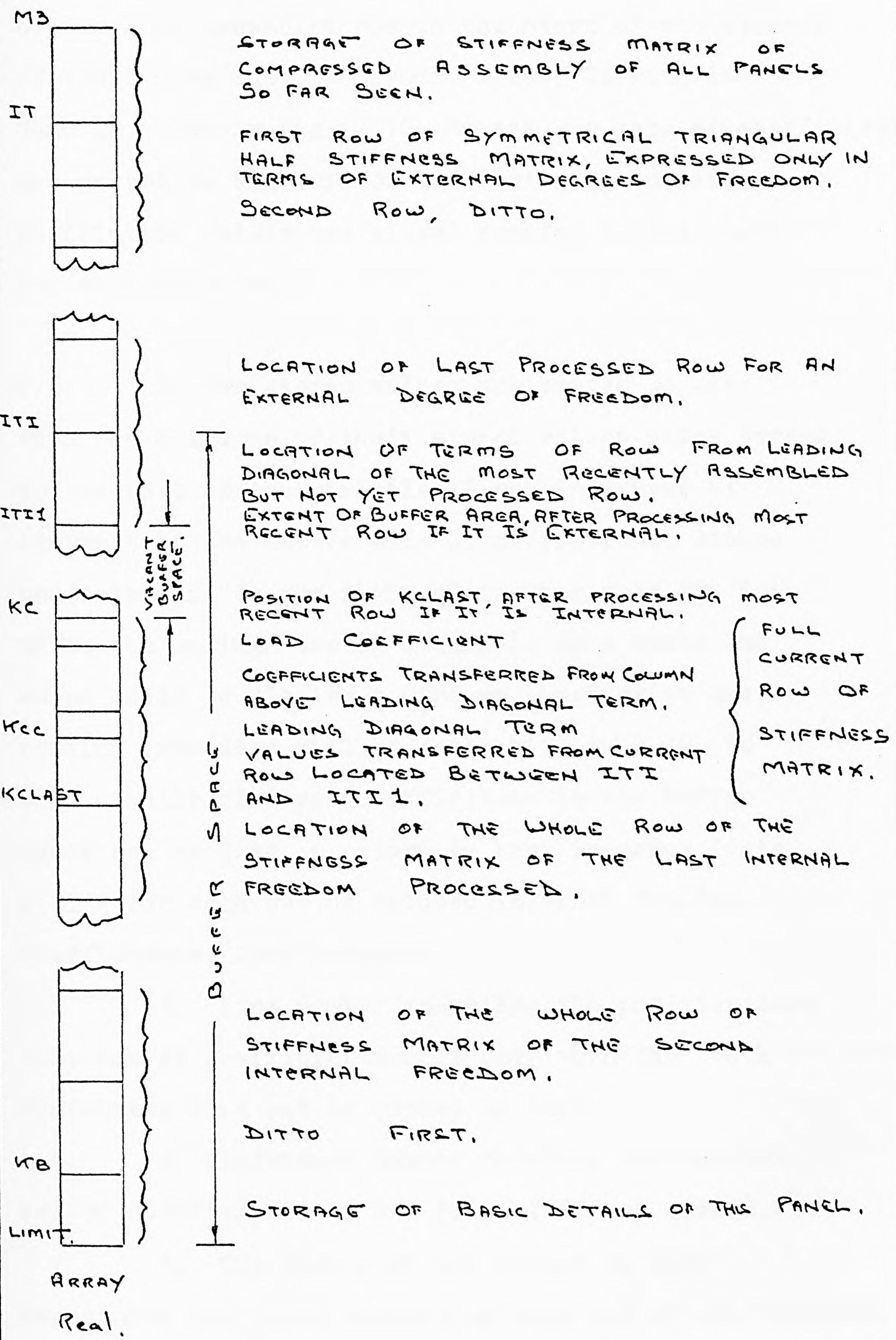
7.8.1 The derivation of a stiffness vector has now been shown. As explained in chapter 5, internal degrees of freedom are continually removed from the matrix. This process is contained in the subroutine REDUCE, which also controls the subroutine MODIFY.

7.8.2 The principal task for REDUCE is to reduce N_1 equations in N_1 unknowns to N_2 equations in N_2 unknowns, without loss of sense. It has been shown by Klyuyev and Kokovkin-Shcherbak (42) that no method using rational operations can for a dense matrix take fewer operations than Gauss elimination. More efficient methods which do not use Gauss like methods are taking advantage of zero elements and symmetry. It was this type of thinking which led to the wavefront concept, by Bamford and Melosh (43). Looking at the probelm we are to solve we see that we are not trying at this stage to solve the equations but only reduce their number. Therefore a selective Gauss forward substitution is used, (selective because only internal rows where they occur are to be used as pivot rows). In effect this is a wavefornt technique.

7.9 Subroutine REDUCE.

7.9.1 The assembly of the panel's stiffness matrix is allowed to proceed line by line until the line formed by the first internal degree of freedom has been completed. REDUCE is now called for the first time to transfer this row to a buffer area and by its calling MODIFY, modify all existing coefficients for the non-appearance in the stiffness matrix of the row and column corresponding to this degree of freedom. Once REDUCE has been called it is called after the assembly of every remaining line of the matrix whether or not it is an internal degree of freedom. As explained later subsequent calls to REDUCE cause it to perform more than the little mentioned here.

7.9.2 After the inversion of the stiffness matrix representing all the remaining external dgrees of freedom of the whole structure a rebuilding process is started. For each panel, DISP puts back into a vector of the displacement values of external freedoms, all the displacements of the internal freedoms.
To achieve this, for each internal freedom a set of coefficients must be found with which to form an equation relating the next internal displacement as specified by the scanning process to all the external displacements and those internal displacements already calculated. It is at the stage of REDUCE when a row corresponding to an internal freedom is formed that these coefficients are set aside. The area set aside forms a buffer space, which expands or contracts to use all the available core, and as more space is needed for the storage of coefficients of rows of external freedoms the buffer area is sacrificed and data stored there is copied onto disc. The requirement at the reconstruction stage is for coefficients for all freedoms that applied at that stage of ASSEMB, to be available. This means that the current row of the triangular stiffness matrix is not sufficient and all those coefficients directly above the leading diagonal term must be copied from their locations into the buffer space just referred to. This buffer space occupies the area of array Real from the end



Relative positions of storages of rows of stiffness matrix.

Fig. 7.10

of the last assembled row to the start of the storage of basic data such as element sizes. Storing is done as shown in Fig. 7.10. Subsequent sets of coefficients are stored to the left of the last set, but indvidual coefficient values are stored running to the right for each equation.

7.9.3 Buffer stored values are copied to disc when the build up of their stored values which extend to the left of the variable KB are in danger of overwriting the active sets of coefficients stored and extending to the right of the variable IT. This makes the maximum use of available core space and keeps calls to disc to a minimum. Copying to and reading from disc will be dealt with in 7.12. To keep a tally of these coefficients in the buffer space and on disc, a record is kept in array Idata giving for each set of removed internal freedom coefficients, four integers:

1. Item number indicating the position that this set of coefficients will have when the block containing this set is copied to disc.

2. Reference number relating the current set of coefficients to the full stiffness matrix.

3. The number of the record on disc containing the block containing this set of coefficients.

4. The number of coefficients contained in this set.

On completion of the assembly of any panel, this block of records is also put onto disc and the space vacated, used for the same purpose for the next panel.

7.9.4 Recovery of the coefficients in the column above the leading diagonal is arranged by a loop in which the subject coefficient's location is found by calculating the number of locations to the end of the current row and then subtracting from it the loop counter value. These values are stored as shown in Fig. 7.10. The variable controlling the start of the storage for the current set of coefficients is found by subtracting the number of coefficients to be stored from the last value it held when the row at that time corresponded to an internal freedon and not just the last equation. It is necessary to make this distinction because the number of coefficients in each full row does not maintain a simple relationship to the number in the previous row. Further, a row of an external freedom is only in the buffer until this next line is formed, but temporary presence is sufficient to confuse the correcting of the controlling variable. If this variable is not kept constantly updated, the stored coefficients may overwrite the first values of the last stored row. Coefficients of the current row starting with the leading diagonal are stored as they are formed in the space to be occupied by the next external row after corrections to its coefficients by REDUCE. Since they are located consecutively they may be easily transferred to occupy the spaces in the buffer immediately behind those values already obtained from the column above the current leading diagonal term. It is necessary to clear the locations after the last processed external row, which have been used to store the current row from the leading diagonal onwards, since the next set of coefficients to occupy this space may not overwrite all the previous non zero values.

7.9.5 The set of coefficients just stored in the buffer area has been formed, as have all the sets of values in the buffer area, by two distinct stages. These are:

1. The recovery from the existing array of the coefficients occurring before the leading diagonal, which have been corrected for the removal of all the previous internal degrees of freedom.

2. The assembly of the leading diagonal term and the remainder of that row, formed without any reference to the effect of the removal of previously seen internal freedoms.

7.9.5.1 On completion of these two stages the reduction process can go no further for any row until the second stage referred to above has been corrected. This correction is required so that though that part of the row contains no cross terms directly related to previously removed internal freedoms, the effect of their removal on the existing coefficients could be made. This process is handled by MODIFY and is described in 7.11.

7.9.5.2 There is an exception to calling MODIFY, in that if the set of coefficients in the buffer is the first set, then MODIFY is not called. This is because since it is the first row related to an internal freedom then its terms cannot be altered since there have been no previous terms removed.

7.9.6 The process used to correct the stiffness matrix rows so far assembled for the effect of removing the row of coefficients of an internal degree of freedom is by application of the first stage of a conventional Gaussian elimination. If the current row for assembling is the N^{th}, then coefficient directly above k (N,N) of the J^{th} row may be removed by multiplying the N^{th} row by k (J,N) / k (N,N) and subtraction it from the J^{th} row, where if J is less than N, k (J,N) is obtained from the N^{th} column. This is repeated for rows 1 to N - 1. At this stage the Nth column apart from k (N,N) is all zeros. All coefficients to the right are now moved over to fill those empty spaces and the N^{th} row all put to zero. The next row, the $(N + 1)^{th}$ of k may now be assembled. This will occupy the space left by row N. Again a full list of coefficients is formed, put away in the buffer and the records updated. Since previously coefficients were moved to the left, k (J,N + 1) is now directly above k (N + 1, N + 1). This (N + 1)th row has not been affected by the removal of its N^{th} coefficient. Reference to the records show what was the number of the coefficient removed on any previous occasion. This coefficient value can then be found from the offloaded details to become a delayed denominator. Also required is the numerator of the current row lying vertically below the coefficient just recovered. The numerator value has not been formed with the current line since it is to the left of the leading diagonal and had it been formed it would not have been in the partially processed state necessary to be compatible with denominator term. Contained in the set of offloaded coefficients is the symmetrical term, though in this case the terms are not equal about the leading diagonal. Since this term is being recovered from the same stored row as the delayed denominator it must be subjected to all the same processing and thus the two values must be compatible. The current $(N + 1)^{th}$ row may now be modified for the removal of its virtual N^{th} coefficient. Such a routine may go on continually using the same row of the partially assembled stiffness matrix until an active degree of freedom is assembled. In this case the full set of coefficeints of remaining degrees of freedom is again assembled, but in this case no records are kept. The coefficients of this active row must however still be modified to allow for the removal of inactive rows of the matrix.

7.9.7 In the case of an internal degree of freedom the whole row in its final form remains in the buffer space in exactly the locations it occupied when it was first put there. Whereas the final form of a row of an external degree of freedom must be moved from the buffer space and added to the end of the partially formed stiffness matrix formed of only external degrees of freedom. Only the terms of the row starting at the leading diagonal are returned, which are the only terms upon which MODIFY acted.

7.9.8 The symmetrical triangular half of the stiffness matrix and load vector are loaded as a single vector with the load term going immediately ahead of its leading diagonal term. Every time an internal row is removed the lengths of the active rows grow to the right so that the last coefficient position lines up with the last coefficient location of the internal, but is subsequently moved to the left filling the zero location caused by the removal of the internal degree of freedom's coefficients. Continuing this theme of the active row growing, consider the assembly of degrees of freedom towards the end of the formation of the stiffness matrix. In this case the active rows will grow past the end of the array before the removal of coefficients and the consequential moving to the left brings the stiffness matrix back to its expected size, there is therefore a small amount of core space required more than would appear necessary at first consideration.

7.10 Subroutine MODIFY.

7.10.1 The purpose of this subroutine is to modify the rows of the stiffness matrix as prepared by ASSEMB into a form which REDUCE can handle. The insertion of this subroutine between the assembly and reduction stages is due to the method of assembly.

7.10.2 Consider assembling a full row of the stiffness matrix for any degree of freedom, whether internal or external but which is assembled after at least one internal freedom has been processed. Some of the terms of this row in locations before its leading diagonal term will be cross product terms related to internal degrees of freedom. As explained in 7.9 as each row of an internal freedom is assembled, REDUCE removes that row and all of its already assembled cross product terms. Therefore this full row contains coefficients of freedoms which as far as the assembled stiffness matrix is concerned do not exist. In the assembly process only the part row from the leading diagonal is actually assembled. This part row is identical to the corresponding part of the full row just described. If now that full row were to be modified to remove those coefficients which are not acceptable to the existing stiffness matrix, then the part of the row from the leading diagonal term would also be changed. It is the function of MODIFY to act upon the part row as assembled and produce a row which is compatible with those freedoms of which the stiffness matrix is so far formed.

7.11 Action of MODIFY.

7.11.1 The buffer area and disc if the buffer area has overflowed containall the coefficients of the rows of the stiffness matrix corresponding to all removed internal degrees of freedom for use later in the derivation of the internal displacement values. This data together with the records held in array Idata may be used to correct the current row as follows.

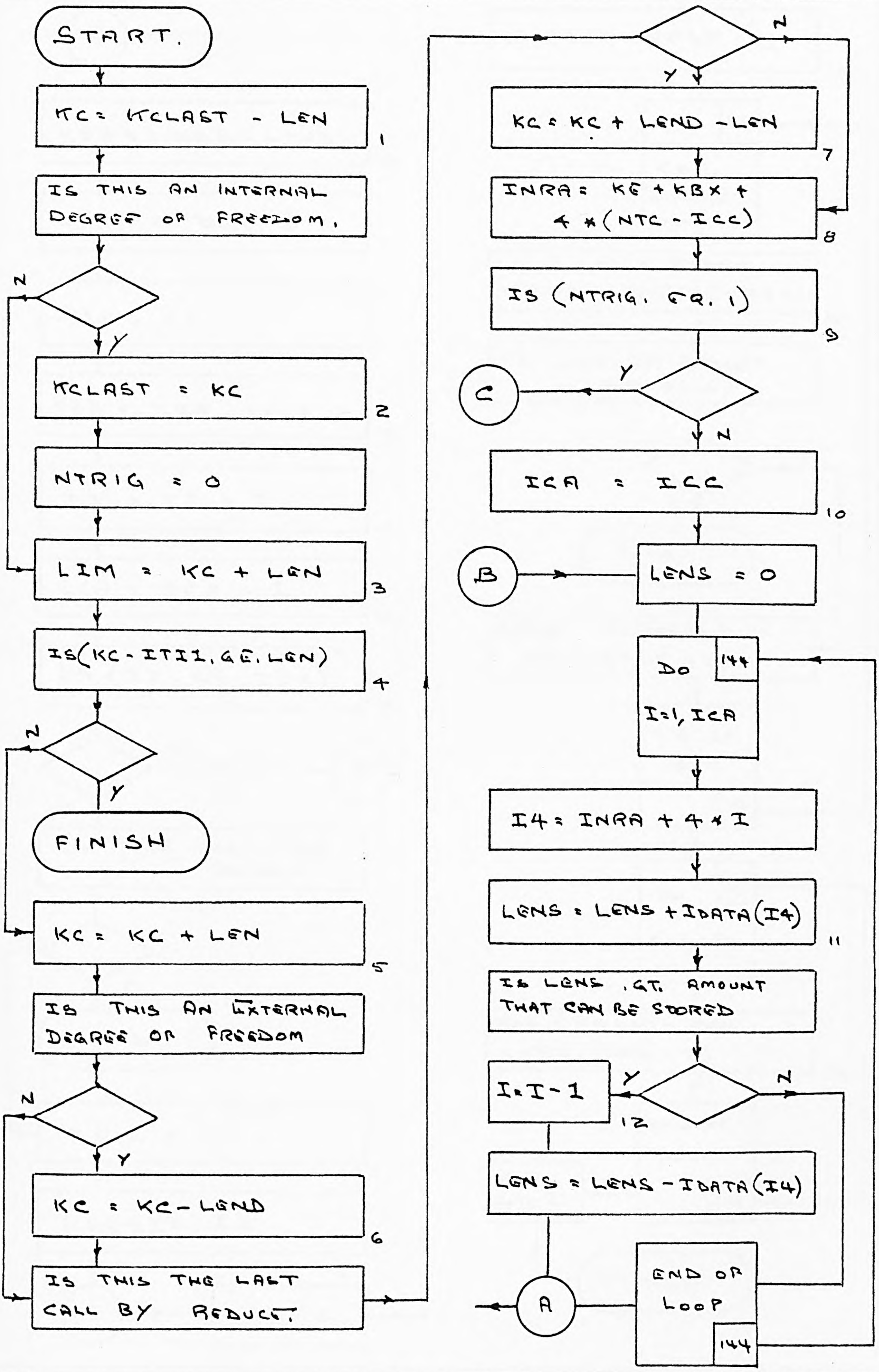
7.11.1.1 Two loops are set up. The outer controls the selection of the stored equation, whilst the inner controls the selection of the coefficient in the row which has just been assembled. The outer loop needs as a limiting value the number of equations which have already been put into the buffer space. This is easily available by reference to the variable NTC, being the number of times REDUCE has been called for an internal degree of freedom. Since if the current equation is for an internal freedom then NTC will include the current equation, but it is the one being operated upon and not the one to be referred to by the outer loop. Then the outer loop counter limit on the number of equation to be seen must be NTC - 1. The current equation may not be an internal freedom, so a variable NADD must be added to NTC - 1, having the value 1 if the current row is external.

7.11.2 Reference to the Idata records for the (Total - I)th set of coefficients stored away will produce the value of the local degree of freedom which was removed at that time and the number of coefficients which its equation occupies in the buffer. Reference is not made to the Ith set of coefficients because it is more efficient to work back starting from the last equation so that if the number of variables that can be changed is less than the limit of the outer loop counter, then a command can be triggered to cause a jump out of the loop.

7.11.3 It may be that during the assembly of the current panel the buffer has been filled and some equations have been put to off disc. If this is so and the jump referred to in the last paragraph has not been actuated then those equations on disc must be brought back to buffer in blocks as large as possible so that the time spent calling to disc is reduced to a minimum. The operation is conducted by RECALL and is described in 7.13.

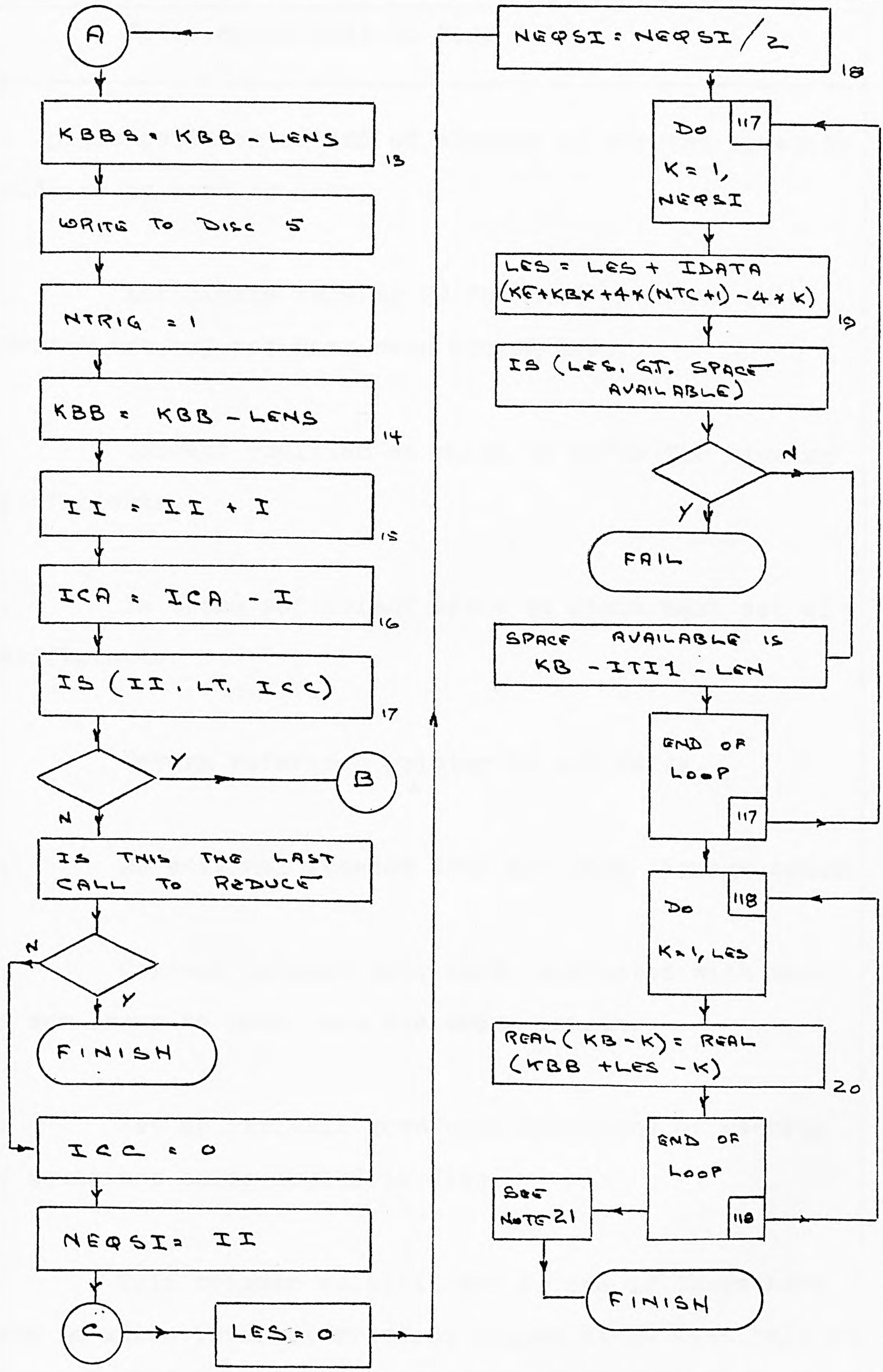
7.11.4 One of the values recovered from the records was the number of coefficients in the equation being referred to. This enables the variable indicating the start of the I^{th} set of coefficients to be continually updated. A multiplier may be formed by the division of one term by another. The numerator of these two terms is the cross product term between the current row and the reference row. The other term is the leading diagonal term that applied when this reference equation was the current equation. In finding these terms account must be taken not only of where does the coefficient storage of this equation start but that these equations were not formed exactly in the order in which they are stored. Due to the non storage in the buffer area of the external freedom equations, the J^{th} stored equation does not have its leading diagonal term in its $(J-K)^{th}$ location, where K is the number of coefficients already removed before this equation was stored. Similarly location problems arise with the cross product term. A measure of the number of missing locations is provided by the difference between the total number missing for the equation just assembled and the number of equations stored between that equation and the I^{th} equation.

7.11.4.1 The inner loop continually operates on the row which has just been assembled by subtracting from the K^{th} term the product of the multiplier just formed by the outer loop and the corresponding term of the equation selected by the outer loop. Once again, allowance must be



Flow chart showing arrangements for triggering reading to disc and updating the associated variables. Part 1

Fig. 7.11(a).



Flow chart showing arrangements for triggering reading to disc and updating the associated variables. Part 2

Fig. 7.11(b).

Notes applicable to Fig. 7.11

1. Update position of end of storage of current equation ready to be read to disc.

2. Anticipate forming KC for next row when this current row may not have been stored here.

3. Correct position at which to refer for storing coefficients.

4. Is there sufficient space to store next set of coefficients.

5. Return reference pointer to old value.

6. An external freedom does not need storage space.

7. Correct pointer previously corrected with what is now known to have been the wrong length.

8. Set up variable governing locations of records of equations being copied to disc.

9. This trigger is still set to one if there have been no extra internal freedoms stored since last call to disc.

10. Temporarily store number of inactive equations formed since last call to disc.

11. Update number of coefficients being copied to disc for I equations.

12. Where length was overestimated by last loop, correct number of equations and length back to permissable values.

13. Form variable being position from which disc is to copy.

14. Set variable indicating to where equations have been removed.

15. Update number of equations put to disc so far, this time.

16. Update number of equations remaining.

17. Are there still equations to go to disc?

18. Set up value of sets of coefficients to be retained in core.

19. Count number of coefficients to be stored if K equations are to be retained.

20. Transfer coefficients from buffer space which is to be emptied to space allocated at other end of buffer.

21. The following group of variables must be reset.

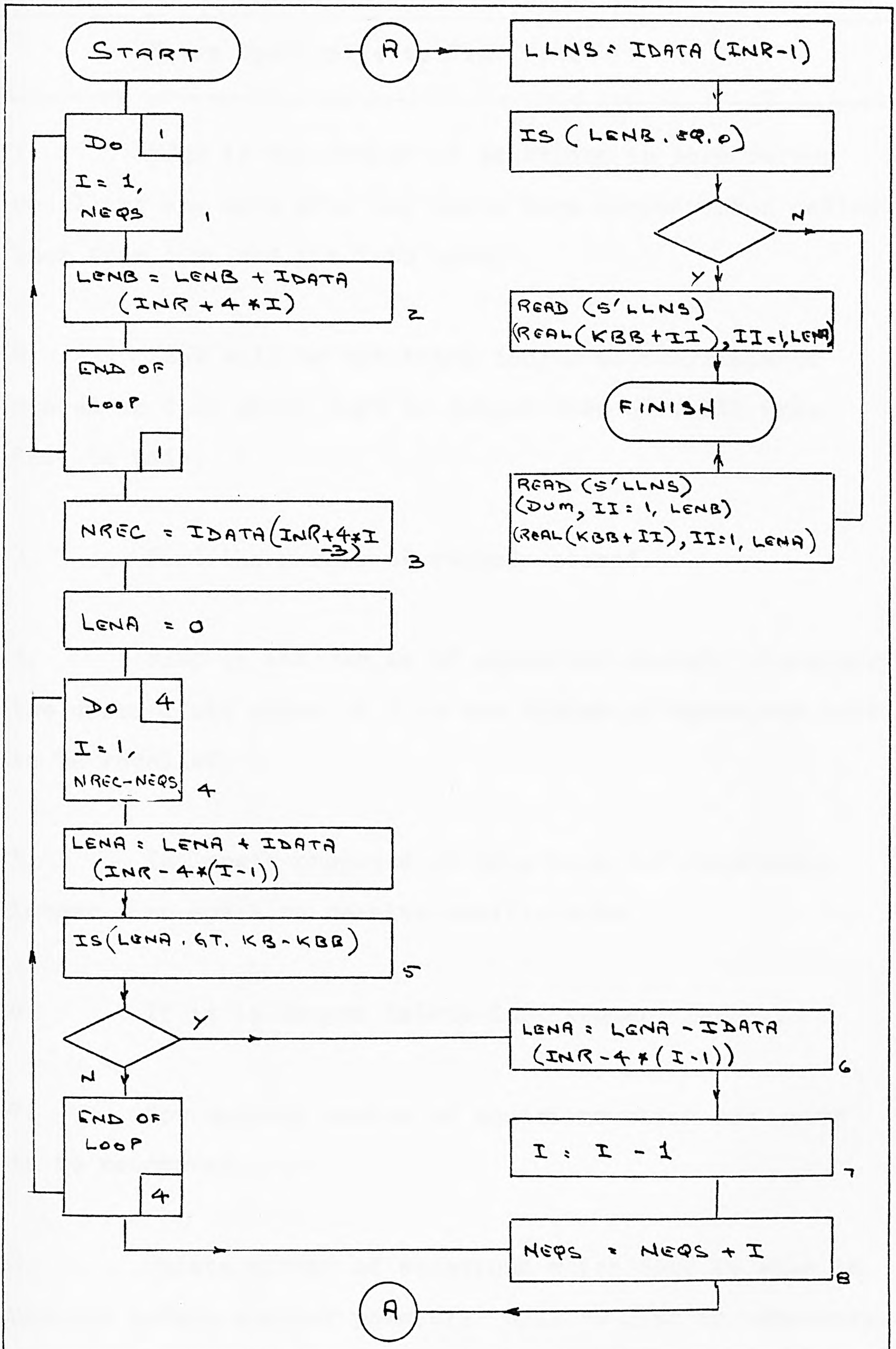
KBB governs to where new coefficients are stored and account must be taken of those old coefficients still retained in core. KC, LIM and KCLAST depend upon the position of KBB. IC is put to zero as new disc record now applies.

made for the missing locations.

7.12 Reading to and from disc.

7.12.1 It can be seen from Fig. 7.10 that REDUCE can operate satisfactorily calling MODIFY as necessary so long as there is sufficient empty buffer space left between the end of the last row of the reduced stiffness matrix at that stage and the start of the last row of the stored for an internal degree of freedom. Sufficient in this context being sufficient to store the next full row of the stiffness matrix of the complete panel, whether this is an internal or external freedom. In the event of this space proving insufficient, the whole of the contents of the buffer space less any part which may have already been copied to disc but is still retained in the buffer is copied onto disc. An additional variable is necessary to keep track of the point separating new sets of coefficients just being stored in the buffer and those which existed in the buffer before any copying to disc was triggered. The position of these coefficients on the disc has already been anticipated by REDUCE and as explained in 7.9 stored in core before any need to call for disc space.

7.12.2 MODIFY uses a process of referring to the most recently stored sets of coefficients first and limiting itself to refer to those to which it is necessary,obviating the need to recall some which have been stored for a comparitively long time. This idea is efficiently assisted now by making available as empty buffer space only about



Flow chart showing the method of finding the length of the record to be recovered and the length to be skipped to reach it.

Fig. 7.12.

Notes applicable to Fig. 7.12

1. NEQS is the number of equations in core before recalling any data plus any which have already been called back from disc and put into core.

2. LENB will be the total length of coefficients stored on disc which must be jumped when a recall from disc is made.

3. Find the number of records stored.

4. NREC is the number of equations stored, therefore the upper limit value of I is the number of equations left to be recalled.

5. Is length proposed at this loop for recalling, longer than space to receive coefficients.

6. If it is longer delete last segment length.

7. Now correct number of equations which are going to be recovered.

8. Update number of equations which core is able to process before another possible call to disc is necessary.

half of what is available and transferring the most recently stored coefficients which were not changed in any way by being copied to disc from the left hand to the right hand buffer space. The vacated buffer space is now available for use as smaller buffer space and extra core space for the next rows of the stiffness matrix. The flow chart at Fig. 7.11 shows the arrangements for transferring complete sets of coefficients, setting up various triggers and revaluing variables when a call to disc is necessary.

7.13 Subroutine RECALL.

7.13.1 In the event of REDUCE having had to put some data onto disc and if that data is required by MODIFY then as shown in Fig. 7.12 subroutine RECALL arranges to bring blocks of coefficients back from disc and places them in the buffer space. Before starting the processing of the next row supplied by ASSEMB, RECALL performs one last duty. That duty is to reload the buffer with the coefficients that were there when the last row arrived. It will be remembered that the buffer is left after copying its contents to disc with the most recently formed sets of coefficients, however these are overwritten by the subsequent action of RECALL. But they are available on disc and since they are the most recently formed and stored on disc, they must be available on the record one number before the current number. Therefore before calling RECALL a variable stored the value (MT - 1), where MT is the current disc record number. After RECALL has finished it can cause one last call to disc to prime the buffer space.

This will only apply to those sets of coefficients stored in the initially halved buffer. Those formed and stored in the vacated area remain undisturbed by RECALL until such time as the previously vacated buffer space is filled in which case, these new sets are copied to disc. Now the previously halved area of buffer is halved again. The program is set to stop if the buffer space becomes too small.

8. Matrix Manipulation.

8.1 Outline

8.1.1 In chapter 5 it was shown how the stiffness matrix for a single stiffened flat plate could be assembled row by row. Later in chapter 7 it was shown that this matrix could be finally evolved as a smaller matrix, but still equivalent to the full stiffness matrix. This matrix was expressed in terms of the degrees of freedom of those nodes situated on the edges of the panel plus any nominated freedoms of nodes away from the edges.

8.1.2 The development of a method is shown in this chapter which allows the most recently thus assembled stiffness matrix equivalent to be selectively modified so that it can be added directly to the stiffness matrix representing all the previously processed panels. A stiffness matrix of this form though occupying only a fraction of the core space required for the full stiffness matrix of the structure will grow with each additional panel's contribution until, if there are sufficient additions, the matrix exceeds the core available. To reduce the possibility of this happening a process of compressing the existing stiffness matrix after each new panel's contribution has been added, is shown in the last part of this chapter.

8.2 Action necessary to facilitate the addition of two matrices.

8.2.1 Consider two structures with different orientations, but having some common nodes, so that the degrees of freedom occurring at those common nodes are not aligned with each other.

8.2.1.1 Let Eq.(8.1) and Eq.(8.2) represent the partitioned load / displacement relationship related to their own local axes, for panels 1 and 2 respectively.

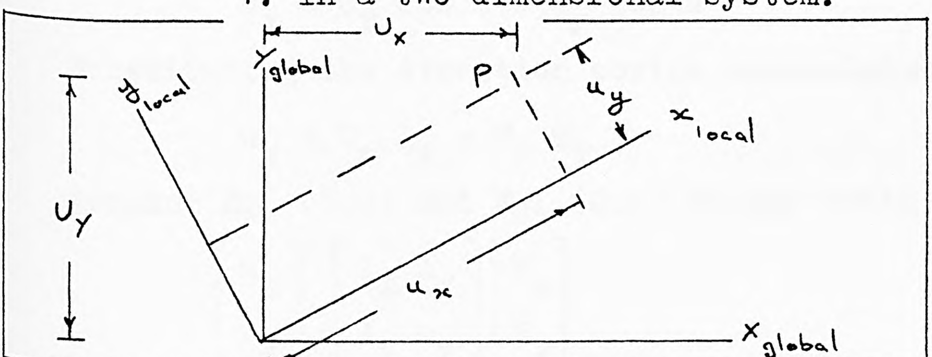
$$\left[\begin{array}{c} p_{1a} \\ \hline p_{1c} \end{array}\right] = \left[\begin{array}{c|c} k_{1aa} & k_{1ca} \\ \hline k_{1ac} & k_{1cc} \end{array}\right] \left[\begin{array}{c} u_{1a} \\ \hline u_{1c} \end{array}\right] \qquad \ldots(8.1)$$

$$\left[\begin{array}{c} p_{2b} \\ \hline p_{2c} \end{array}\right] = \left[\begin{array}{c|c} k_{2bb} & k_{2cb} \\ \hline k_{2bc} & k_{2cc} \end{array}\right] \left[\begin{array}{c} u_{2b} \\ \hline u_{2c} \end{array}\right] \qquad \ldots(8.2)$$

Where the partitioning of the panels is between those degrees of freedom which are unique to that panel and those degrees of freedom sharing a common node.

8.2.2 Relationship between local axes and global axes displacements.

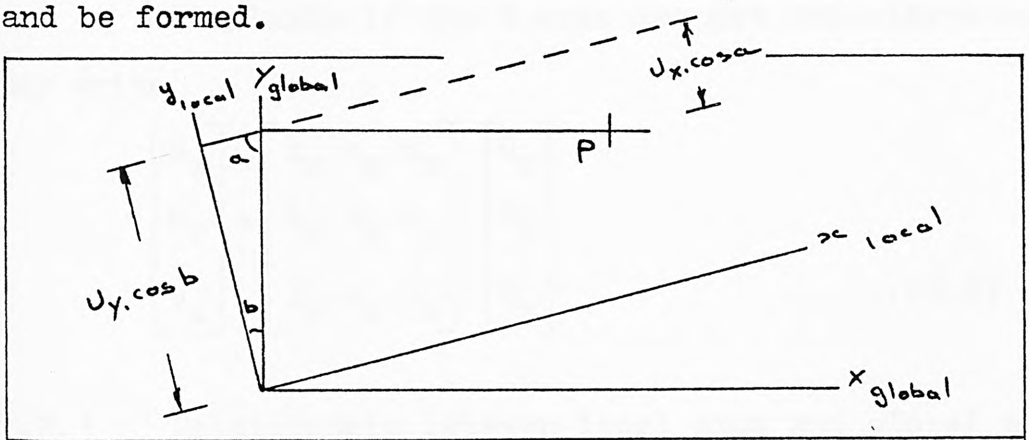
1. In a two dimensional system.



Global / local axes relationship

Fig. 8.1

By extracting details from Fig. 8.1, Fig. 8.2 and Fig. 8.3 and be formed.



Obtaining the y co-ordinate.

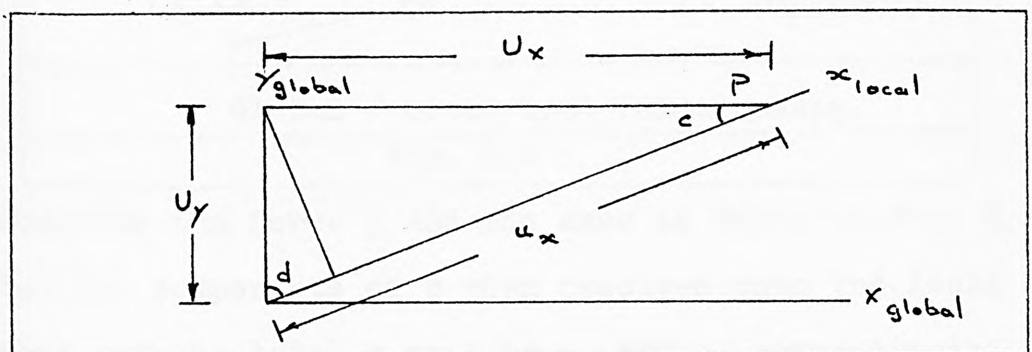
Fig. 8.2

From Fig. 8.2

$$u_y = - U_x \cos a + U_y \cos b$$

Substituting the direction cosine nomenclature

$$u_y = U_y m_y + U_x l_y \qquad \ldots(8.3)$$



Obataining the x co-ordinate.

Fig. 8.3

From Fig. 8.3

$$u_x = U_x \cos c + U_y \cos d$$

Substituting the direction cosine nomenclature

$$u_x = U_x l_x + U_y m_x \qquad \ldots(8.4)$$

Summing Eq. (8.3) and Eq. (8.4) we may write

$$\begin{bmatrix} u_x \\ u_y \end{bmatrix} = \begin{bmatrix} l_x & m_x \\ l_y & m_y \end{bmatrix} \begin{bmatrix} U_x \\ U_y \end{bmatrix} \qquad \ldots(8.5)$$

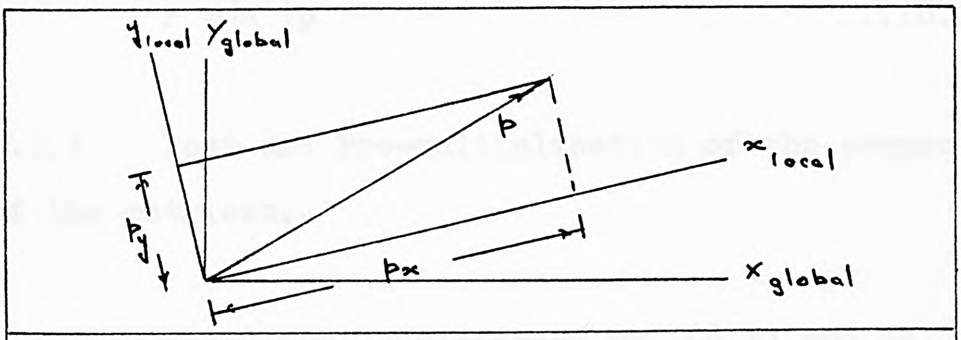
2. In a three dimensional system.

Similarly if the Z axes are not coincident we may write

$$\begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} = \begin{bmatrix} l_x & m_x & n_x \\ l_y & m_y & n_y \\ l_z & m_z & n_z \end{bmatrix} \begin{bmatrix} U_x \\ U_y \\ U_z \end{bmatrix} \qquad \ldots(8.6)$$

8.2.3 Relationship between local axes and global axes loads.

1. In a two dimensional system.



Global / local load relationship.

Fig. 8.4

Consider the force p and the axes as shown in Fig. 8.4. Let the components of p when resolved onto the local x axis and the local y axis be p_x and p_y respectively. Resolving p_x and p_y onto the global axes

$$P_x = p_x\, l_x + p_y\, l_y \qquad \ldots(8.7)$$

$$P_y = p_y\, m_y + p_x\, m_x \qquad \ldots(8.8)$$

Summing Eq. (8.7) and Eq. (8.8) we may write

$$\begin{bmatrix} P_x \\ P_y \end{bmatrix} = \begin{bmatrix} l_x & l_y \\ m_x & m_y \end{bmatrix} \begin{bmatrix} p_x \\ p_y \end{bmatrix} \qquad \ldots(8.9)$$

2. In a three dimensional system.

By similar resolution of componenets we may obtain

$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} l_x & l_y & l_z \\ m_x & m_y & m_z \\ n_x & n_y & n_z \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \qquad ...(8.10)$$

Eq. (8.6) may be written as

$$u = \left[a\right] U \qquad ...(8.11)$$

and Eq. (8.10) may be written as

$$P = \left[a^T\right] p \qquad ...(8.12)$$

8.2.4 Post and Pre-multiplication of the common parts of the matrices.

8.2.4.1 The direct addition of Eq. (8.1) and Eq. (8.2) is prevented because p_1 and u_{1_c} have different orientations to p_2 and u_{2_c}. Expanding the upper row of Eq. (8.1) we obtain

$$p_{1_a} = k_{1_{aa}} u_{1_a} + k_{1_{ca}} u_{1_c}$$

Substituting Eq. (8.11) as it applies to panel 1 we obtain

$$p_{1_1} = k_{1_{aa}} u_{1_a} + k_{1_{ca}} a_1 U_c$$

Similarly this may be applied to Eq. (8.2) and the remainder of Eq. (8.1). Expanding the lower row of Eq. (8.1) as modified by post multiplying by a_1 we obtain

$$p_{1_c} = k_{1_{ac}} u_{1_a} + k_{1_{cc}} a_1 U_c$$

comparing the left hand side of this equation with Eq. (8.12), we see that we can now write

$$P_1 = a_1^T p_{1_c} = a_1^T k_{1_{ac}} u_{1_a} + a_1^T k_{1_{cc}} a_1 U_c$$

Therefore Eq. (8.1) may be written as

$$\left[\begin{array}{c} p_{1_a} \\ \hline a_1^T p_{1_a} \end{array}\right] \left[\begin{array}{c|c} k_{1_{aa}} & k_{1_{ca}} a_1 \\ \hline a_1^T k_{1_{ac}} & a_1^T k_{1_{cc}} a_1 \end{array}\right] \left[\begin{array}{c} u_{1_a} \\ \hline U_c \end{array}\right] \qquad \ldots(8.13)$$

Similarly Eq. (8.2) becomes

$$\left[\begin{array}{c} p_{2_b} \\ \hline a_2^T p_{2_c} \end{array}\right] \left[\begin{array}{c|c} k_{2_{bb}} & k_{2_{cb}} a_2 \\ \hline a_2^T k_{2_{bc}} & a_2^T k_{2_{cc}} a_2 \end{array}\right] \left[\begin{array}{c} u_{2_b} \\ \hline U_c \end{array}\right] \qquad \ldots(8.14)$$

8.2.4.2 What has happened now is that the local stiffness matrices have been selectively changed so that coefficients related to certain degrees of freedom have been referred to global axes so that those degrees of freedom are compatible with others on adjacent panels sharing the same node, whilst the disturbance to other coefficients is minimized.

8.2.4.3 Since now all load and displacement terms are either fully common having the same orientation or the terms in one matrix do not exist at all in the other matrix, the two equations may be added directly.

8.2.4.4 In the case being considered the matrix is not the conventional local stiffness matrix but the matrix in terms of external nodes. This makes no difference since the relationship is still written in terms of the loads and displacements for specific degrees of freedom and it is the loads and displacements that are to be changed.

8.3 Subroutine PARTIT.

8.3.1 The need for PARTIT.

8.3.1.1 It would be sufficient to apply a pre multiplication by $[a]^T$ and a post multiplication by $[a]$ to every coefficient of the stiffness matrix but this would cause an unnecessary growth in the core space occupied and would add to the run time. To keep this operation to a minimum resort must be made to the selective multiplication as outlined in 8.2. Rows and columns of the stiffness matrix are left in the order in which they were at the end of ASSEMB. No attempt is made to group rows and columns. Instead PARTIT prepares a guidance list for the action to be taken by TRANS, which performs the necessary matrix multiplication to the implied partitioned coefficients.

8.3.2 Integers assigned by PARTIT.

VALUE	MEANING
1	Single rotational freedom needing transforming through 180° to make it suitable to add to a rotational freedom having a PARTIT assigned value of 6.
2	Two translation or two rotational freedoms needing no transformation.
3	Three translational freedoms needing no transformation.

4	Initially two translational freedoms but due to being on an edge which is common to a panel inclined to its own panel, needing to be transformed to global reference axes and have a void row and column created to accommodate a third degree of freedom.
5	Three translation freedoms requiring to be transformed to global axes.
6	A single rotational degree of freedom needing no transformation.
N where N is greater than 6.	This indicates a set of N - 6 psuedo external freedoms i.e. internal restraints and / or internal spring endings, occurring between two consecutive genuine external freedoms. Genuine internal freedoms may be situated in a set of pseudo external freedoms without upsetting the counting, though not included in the count.

List of Meaning of the Integers Assigned by PARTIT.

Fig. 8.5

8.3.2.1 This guidance list consists of a set of integers having the values 1 to 6 inclusive plus a seventh value which is greater than 6 whose actual value depends upon the number of internal restraints and spring endings occurring between two consecutive external degrees of freedom. Fig. 8.5 lists the meaning of the values assigned by PARTIT.

8.3.2.2 With these values, translation freedoms at a node are treated completely separately from the rotational freedoms which may be at the same node. Thus a node at which a stiffener is present is represented by two integers, whereas if there is no stiffener only one integer is necessary.

8.3.3 Initial action of PARTIT.

8.3.3.1 For the fast preparation of a list such as that shown if Fig. 8.5, it must be easily known, for any side which is being examined

1. If there is an edge of an adjacent panel common with the current edge.

2. If there is an adjacent panel having the current edge as a common edge, then whether these two panels are in the same plane.

3. If there is an adjacent panel, will the rotational degrees of freedom on each panel, for a common node at a stiffener crossing the current edge be suitably aligned for direct summation?

8.3.3.2 After initializing various variables the opening operation of PARTIT is to set up the first four locations of a five location array A. To provide answers to these questions for PARTIT a loop examines the input data provided regarding the presence and slope of adjacent panels. The absence or lack of relative slope of an adjacent panel on the I^{th} side will not cause A(I) to be filled. If the I^{th} side has an adjacent side with relative slope or the two panels are not suitably aligned so that the common rotational freedoms require some transformation operation then A(I) takes on a unity value.

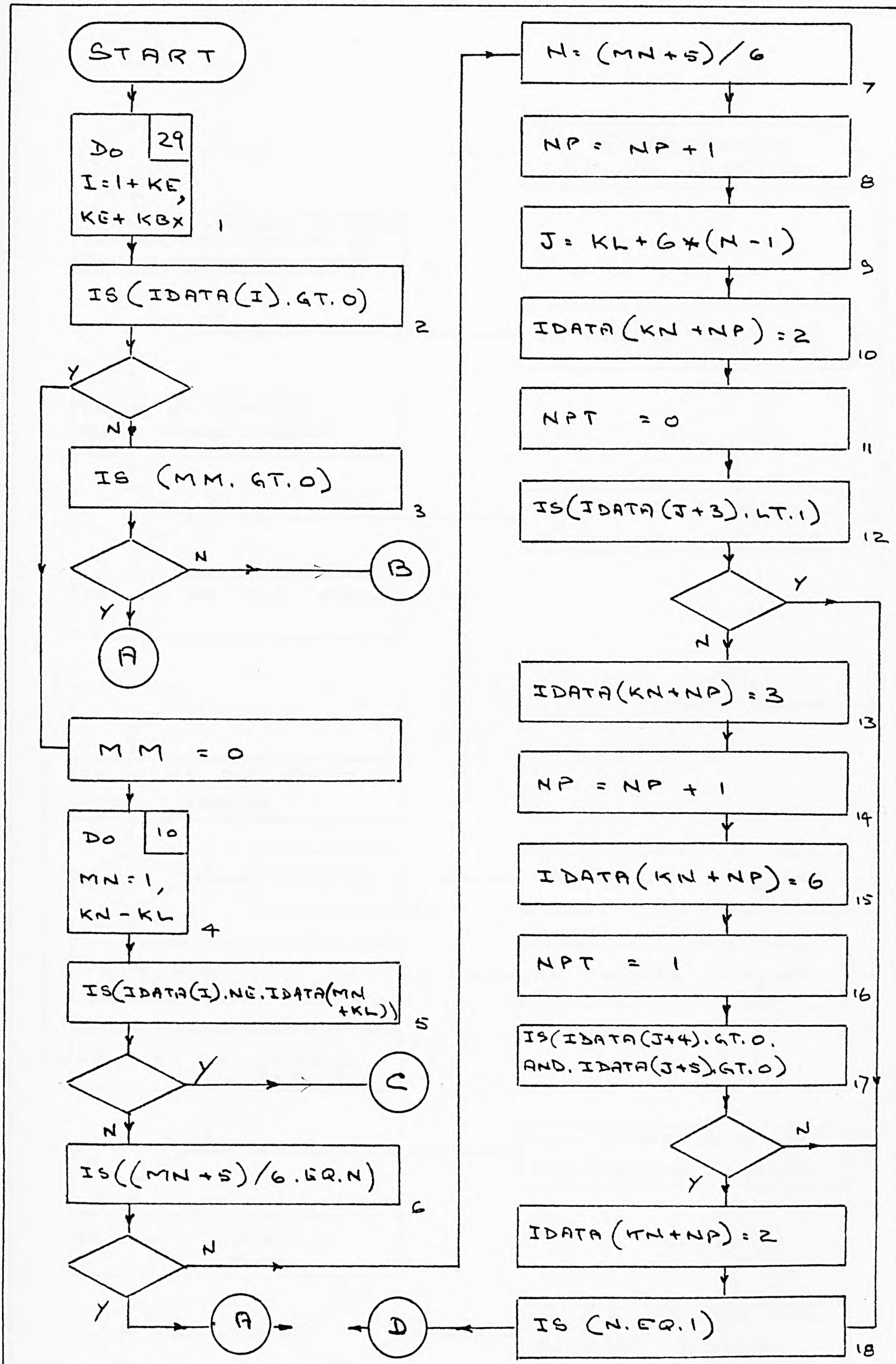
8.3.3.3 The decision as to whether the stiffener crossing the common edge will require transformation for one of the rotational freedoms is made during EDGE. If when assigning reference numbers to freedoms along an edge by copying those already allocated to an adjacent panel sharing this edge, it is necessary to copy them in the reverse order to that in which they occurred when they were first allocated, then the rotational freedoms along this edge must be corrected. This logic can be reasoned as follows. Since the allocation of reference numbers proceeds away from the local axes and as the direction of a rotational freedom is dependent upon the alignment of the axis, if another panel reverses the order of the degrees of freedom, it must be because the local exis for that side is running in an opposite direction. Therefore the rotational freedom must be of an opposite hand.

8.3.3.4 This decision is transferred from EDGE to PARTIT by the storage of four triggers put into Idata.

8.3.3.5 If this is a triangular panel the decision for the third side is copied into the fourth location, since as far as possible triangular panels are treated as rectangular panels and the fourth location is where PARTIT expects to find details of the last side.

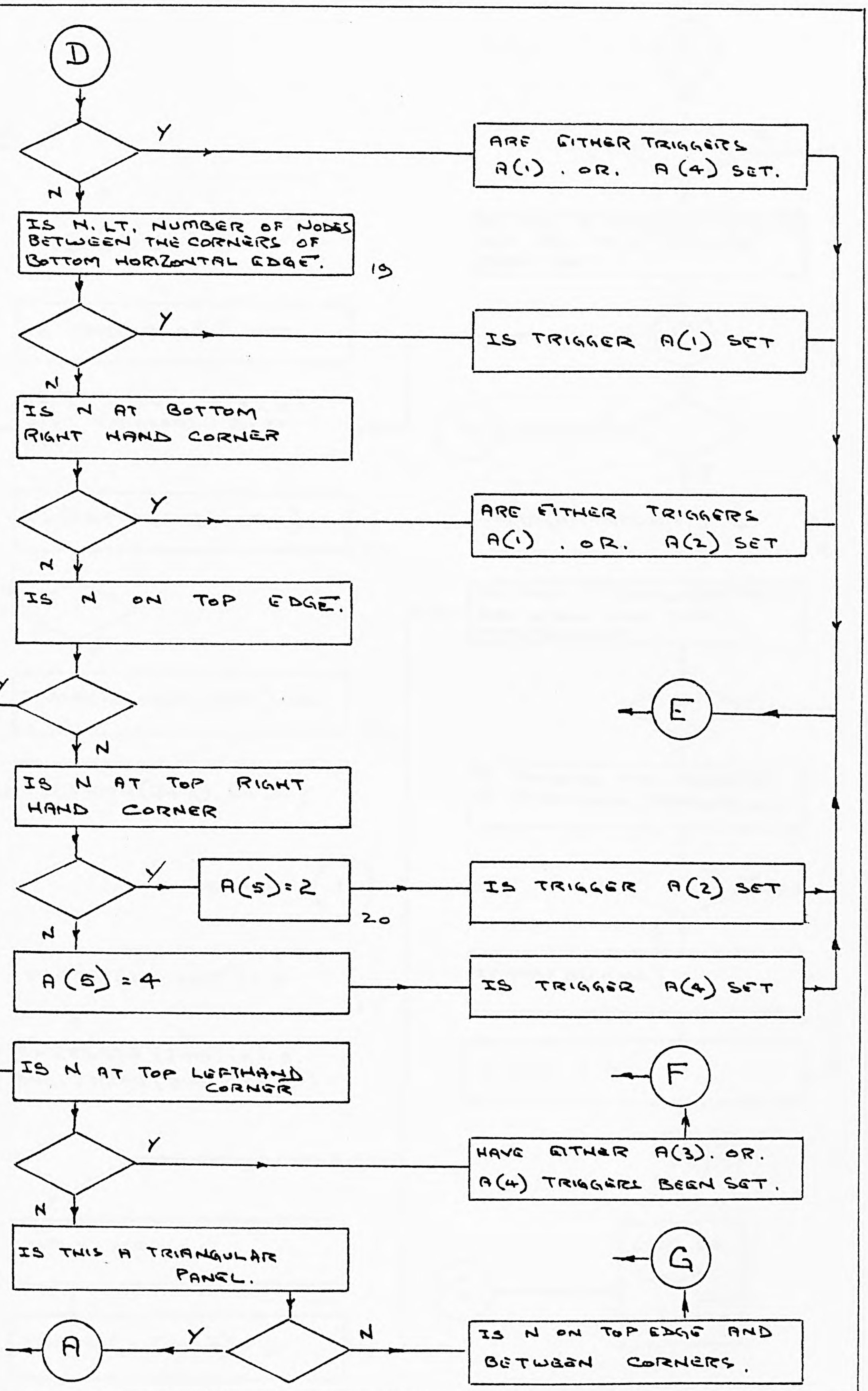
8.3.4 Generating integers by PARTIT.

8.3.4.1 A list of all the degrees of freedom that the current panel has or will have after assembling into its surrounding structure is available in the KE list held in Idata. This list is a consecutive list of freedom numbers and does not contain any empty locations. The KE list does not have every value examined. In the case of degrees of freedom of an external node, only that node's first value is examined. This value is compared with the KL list. The KL list has six locations for every node irrespective of the number of freedoms at that node. Having found the applicable node by the comparison with the KE list, all the remaining freedoms at that node are examined using only the KL list. In this way knowledge is maintained of when a particular node exhausts all its degrees of freedom being offered for consideration. When this occurs and the first value of the next node is needed, the arrangements are made for the pointer of the KE list to skip the remainder not seen at the KE list and move to the next node. It is necessary to examine two lists at once since there may be internal restraints or springs ending on internal degrees of freedom. In both of these events their presence is indicated in the KE list by a negative sign. This sign is necessary to trigger subsequently their removal from the stiffness matrix before assembling the next panel. However the KE list does not indicate which freedoms are grouped together at a particular node, for this reference to the KL list is required but this list contains no indication of internal freedoms needing to be given an integer value by PARTIT.



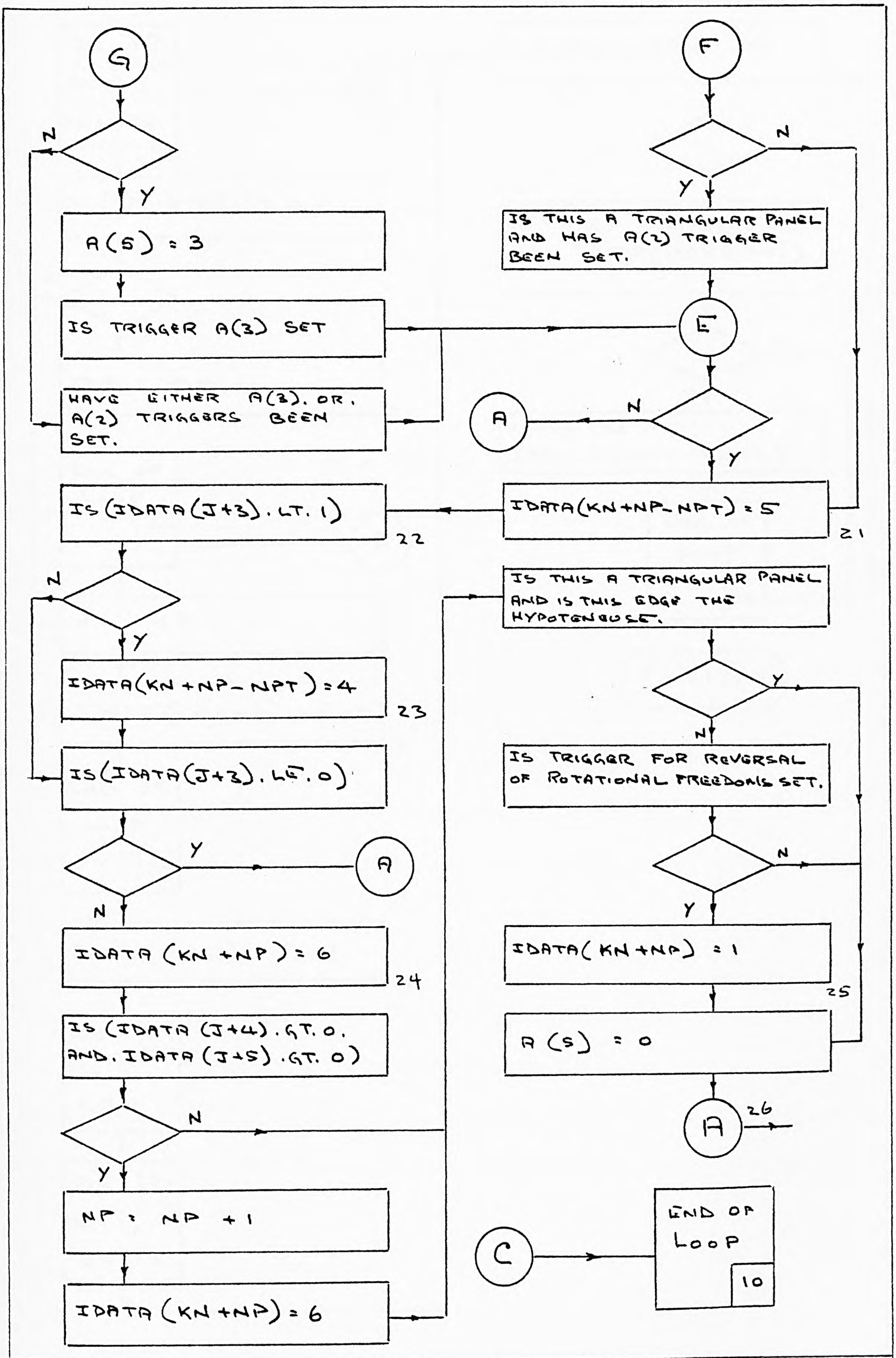
Flow chart for subroutine PARTIT, showing the allocation of integers. Part 1.

Fig. 8.6(a)



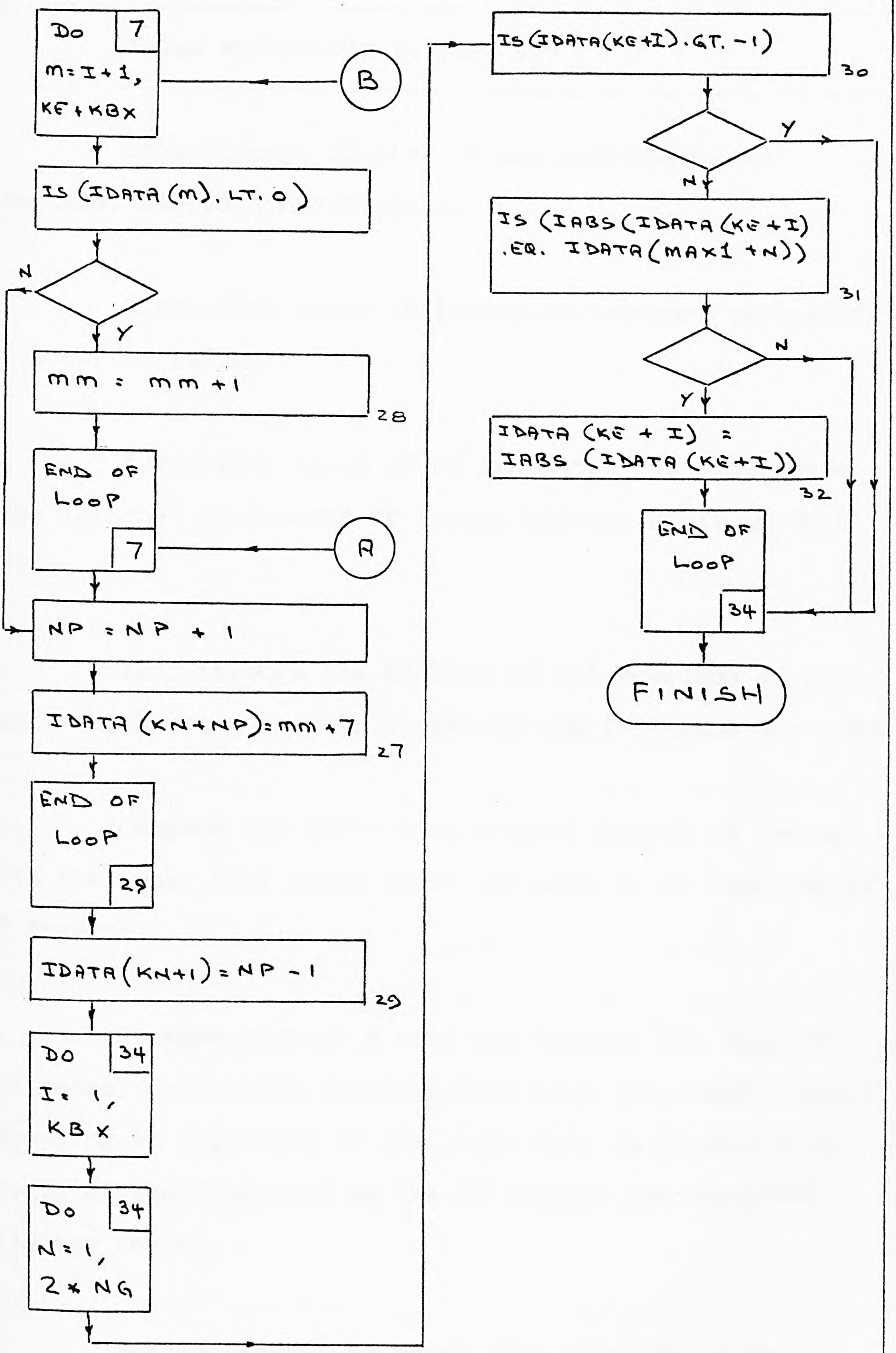
Flow chart for subroutine PARTIT, showing the allocation of integers. Part 2.

Fig. 8.6(b)



Flow chart for subroutine PARTIT, showing the allocation of integers. Part 3.

Fig. 8.6(c)



Flow chart for subroutine PARTIT, showing the allocation of integers. Part4.

Fig. 8.6(d)

Notes applicable to Fig. 8.6

1. Loop through KE list of all current panels' freedoms, stored consecutively.

2. A negative value indicates an internal restraint or a spring ending.

3. A positive value of MM indicates that there are more internal restraints or spring endings still in this set.

4. Loop through the KL list of all freedoms of this panel which are stored in blocks of six locations per node.

5. Compare the outer loop subject degree of freedom with the inner loop value which can only be an edge degree of freedom.

6. Compare current N with the trigger set when MN was reset. Subsequent freedoms have been processed without returning to the start of the loop. This is in effect an update of the I counter as the MN counter has advanced with the search.

7. Set up trigger to cause jump from this node.

8. Update counter of KN list.

9. Locate position of start of block of freedoms of

current node.

10. The minimum that can apply at a node is a two degree of freedom plate node which is not processed.

11. Zero the trigger for correcting the NP counter.

12. Skip if there is no stiffener present.

13. Overwrite the 2 value already allocated since it is apparent that there is a third order freedom at this node.

14. Update count because rotational trigger operates on a different block of the stiffness matrix to translation triggers.

15. The minimum that can apply here is a single rotational freedom, therefore from Fig. 8.5 a value of 6 must be given.

16. Set the trigger to indicate NP counter has moved on one position for the rotational freedom.

17. As two stiffeners intersect a single freedom value is not appropriate, therefore the 6 value must be overwritten.

18. N is a count of the edge nodes processed since the start and includes current node. It does not include

the psuedo external freedoms. This section converts the counting made from the scanning of the panel from side to side to a recognition of the edge currently being considered.

19. Set up reference of which side is current.

20. N must now be at the top right hand corner.

21. NP counter is set now on rotational freedoms. NPT is needed to correct this back to translational position. It is assumed that there is a stiffener present then there must already be a treble group of freedoms needing transforming.

22. If IDATA (J + 3) is less than 1 then the third degree of freedom must be as a result of an adjacent inclined panel and not due to a stiffener.

23. Since there is no stiffener present overwrite the 5 value already given.

24. If there is a stiffener present set up a rotational freedom value for a single freedom.

25. Overwrite the 6 value since this is a freedom to be moved through 180^{o}.

26. The number of internals to be considered between the two edge freedoms is now known, therefore the loop may be left.

27. Increase the value to go in the integer list by 7 to avoid confusion during TRANS.

28. Count number of internal restraints or spring endings before the next external freedom.

29. Store the number of blocks into which the stiffness matrix will be partitioned.

30. Occurring at freedoms of internal nodes spring endings have been stored as restraints. Their presence is indicated by a negative sign. Freedoms having a negative value will now be removed from the stiffness matrix. To retain the springs change their values from negative to positive.

31. Search the spring end list.

32. Correct the value found on this panel's index which is also found in the index of spring endings.

End of notes for Fig. 8.6.

8.3.4.2 If when after processing a node the next examination of the KE list discovers a negative sign, then it is known that this particular freedom is either an internal restraint or the end of a spring. In both cases there is no need to transform such a freedom as it can not be added to any other freedom. Having discovered one negative value the KE list is scanned onwards from that location and a variable MM sums the number of such negatively numbered freedoms occurring before the next positive value. The final value of MM plus 6 gives the value N of the list in Fig. 8.5. The MM value may not become directly the N value since any values less than 7 would be confused with the other integers.

8.3.4.3 The discovery that the current KE list value is positive results in a series of examinations of the set of values stored in the KL list having the same value in the first location of that set as the current KE list's value. The flow chart at Fig. 8.6 shows these examinations. As progress is made through the two lists a variable N is kept as a counter, so that a comparison made with N will indicate which corner or side applies. The fifth location of the vector A is used to transfer the side number to other parts of PARTIT.

8.3.5 Final stage of PARTIT.

8.3.5.1 After completion of the list of integers, the function of the negative signs given to the reference numbers of degrees of freedom being spring endings is complete. These signs must now be removed so that they may be differentiated from those freedoms which are restrained. This is arranged by comparing the absolute values of all the freedoms of the KE list having a value of less than 1 with the list of spring endings which is stored in the MAX1 list. When a match is found the KE list is corrected. For further core savings the list of integers is written over the list of local degrees of freedom, which is no longer required since the panel's stiffness matrix equivalent has been assembled.

8.3.5.2 The final action is to transfer the KE list to follow immediately behind the integer list. This movement allows ease in the action of copying to disc.

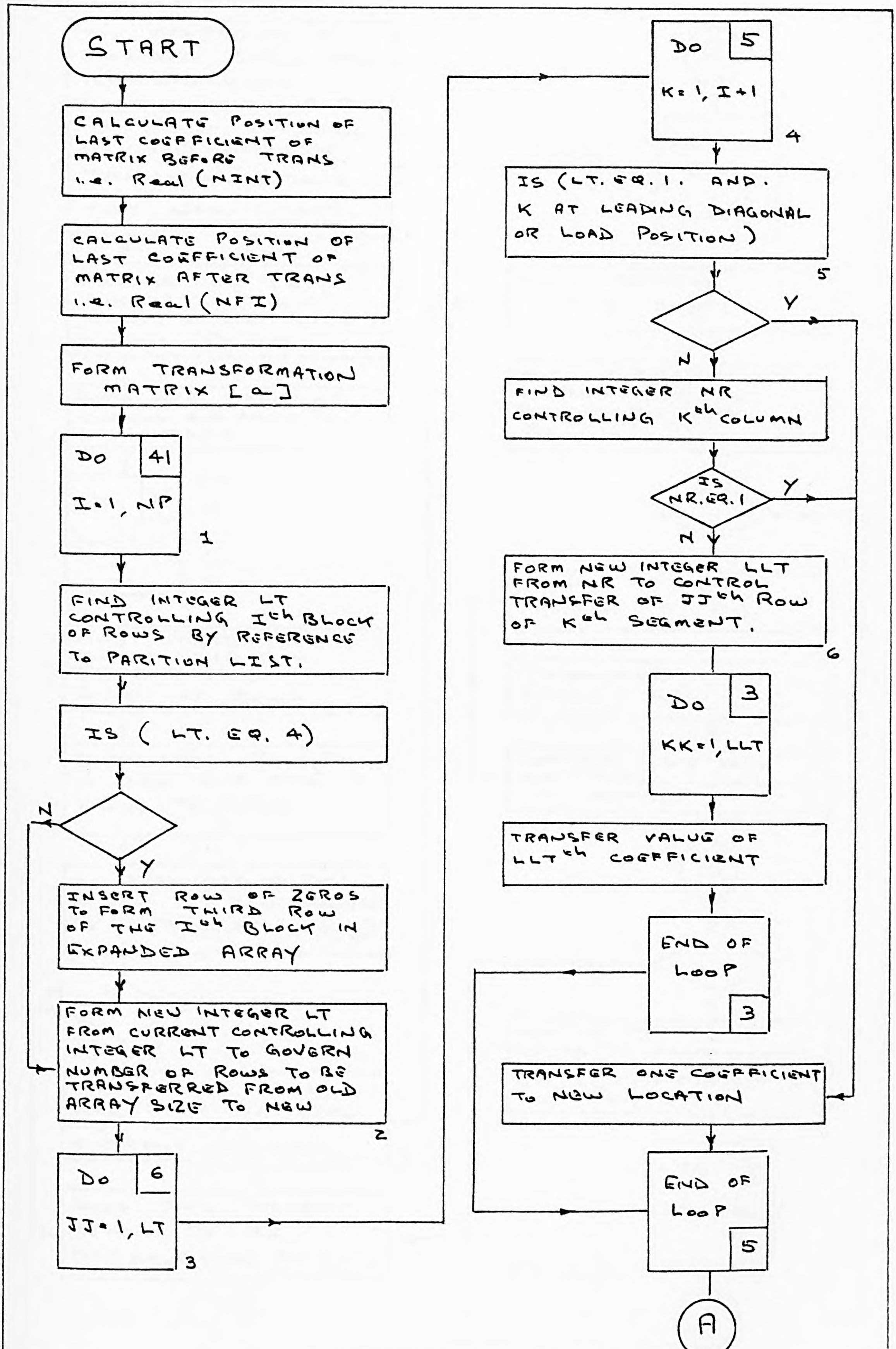
8.4 Subroutine TRANS.

8.4.1 The need for TRANS.

8.4.1.2 Having prepared the list of group integer values, TRANS guided by that list may now be employed to transform the stiffness matrix.

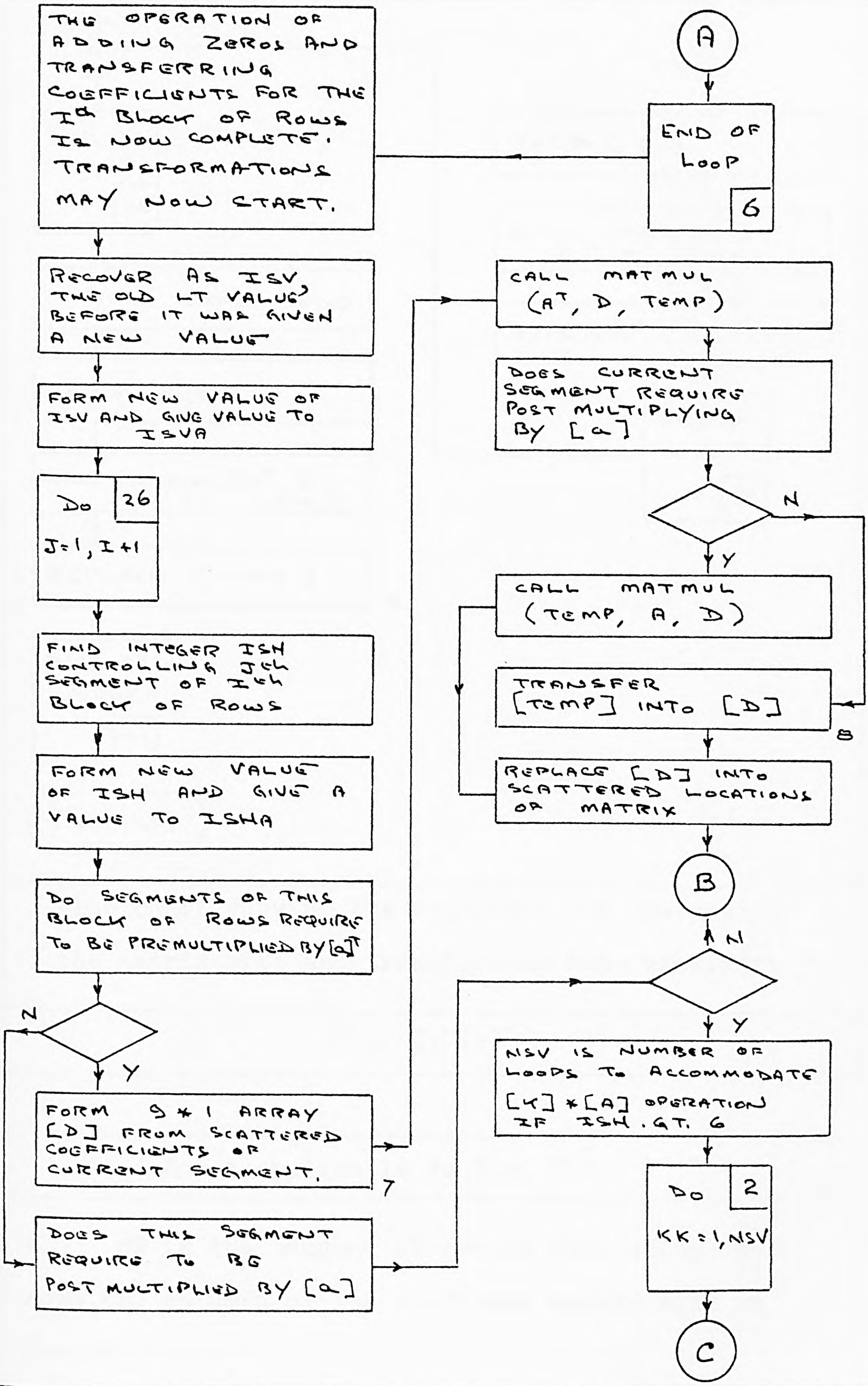
8.4.2 Action of TRANS.

8.4.2.1 The steps of this process are as shown in the flow chart at Fig. 8.7. These are briefly to expand the matrix size whilst totally overwriting its original area thus minimising the extra core required for the expansion. Void spaces are arranged to appear where new rows and columns are to occur. At the same time the coefficients are moved to sites within the final matrix size from which together with the created voids cause them to be in their correct locations prior to transformation. They can then be operated upon directly by transforming matrices not bigger than 3 * 3. The integer grouping values are used to divide the rows into a series of blocks of rows and similarly blocks of rows into segments. Operations start



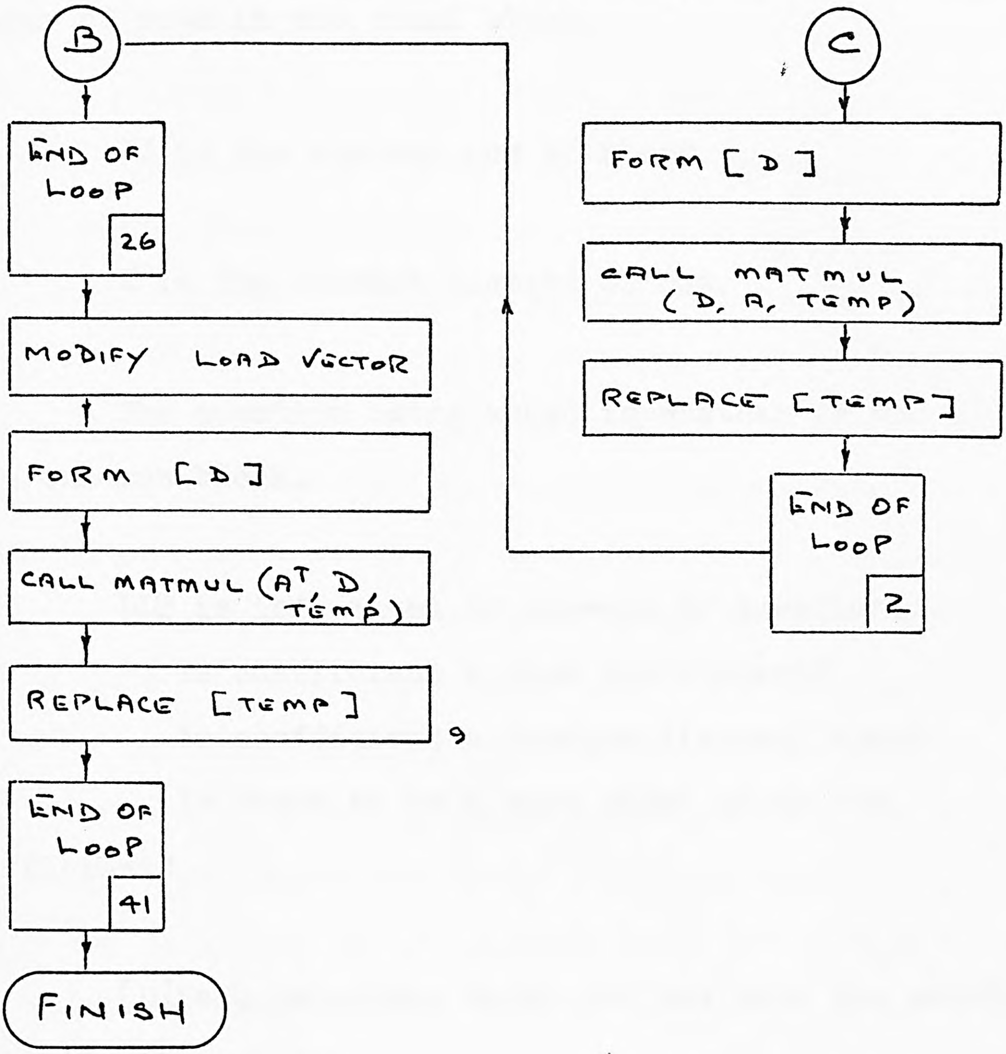
Flow chart showing the technique for inreasing the matrix size and transforming some axes. Part 1

Fig. 8.7(a)



Flow chart showing the technique for increasing the matrix size and transforming some axes. Part 2

Fig. 8.7(b)



Flow chart showing the technique for increasing the matrix size and transforming some axes. Part 3

Fig. 8.7(c)

Notes applicable to Fig. 8.7

1. NP is the number of groups into which the rows and columns of the stiffness matrix will be formed.

2. The number of rows to be transferred is influenced by whether a zero row was added and the

number of rows in the final block.

3. JJ is the current row of block.

4. K is the current segment of row.

5. The question being asked is whether or not this is a one-row block.

6. LLT is influenced by answers to questions:-

Is coefficient a load coefficient?

Is coefficient a leading diagonal term?

Is there to be a zero added after the coefficient?

7. [D]is a temporary array for use with the matrix multiplication routine.

8. Routine is pre-arranged to expect the coefficients which are to be returned to the stiffness matrix to be presented in temporary array [D], when this is not so there must be an initial transfer to [D].

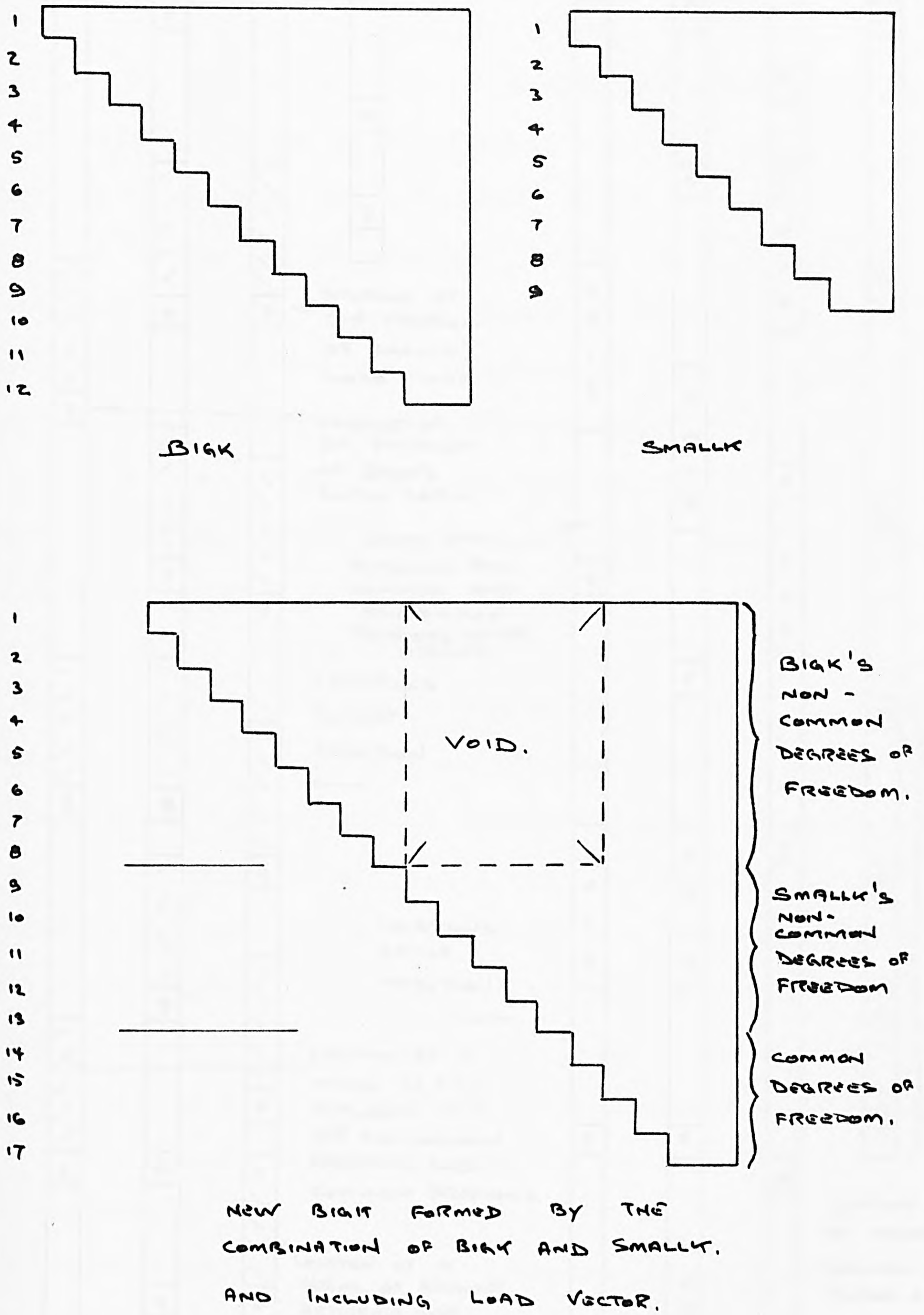
9. Put correctly orientated segment of the load vector back into the array Real.

from the right hand end of the one dimensional stiffness matrix, working sequentially through the segments, this is to prevent premature overwriting of the existing array. Each segment haseffectively two co-ordinates derived from the integer grouping values. One controlling the action to beapplied to the block of rows and the other controlling the block of columns. At each new segment, reference is made to the controlling co-ordinates and a matrix multiplication is performed as required. As an example of this control, a segment having a column value of 4 and a row value of 5 would cause effectively a column of three zeros to be added at the right hand end of a segment 2 * 3 to make it a 3 * 3. Since it is not laid out as a two dimensional array the three zeros are actually scattered as is the whole segment. This 3 * 3 segment of the stiffness matrix would then be subjected to an a^T k a transformation, whereas a segment with the co-ordinates (3,2), for example, would be unchanged.

8.5 Addition of matrices.

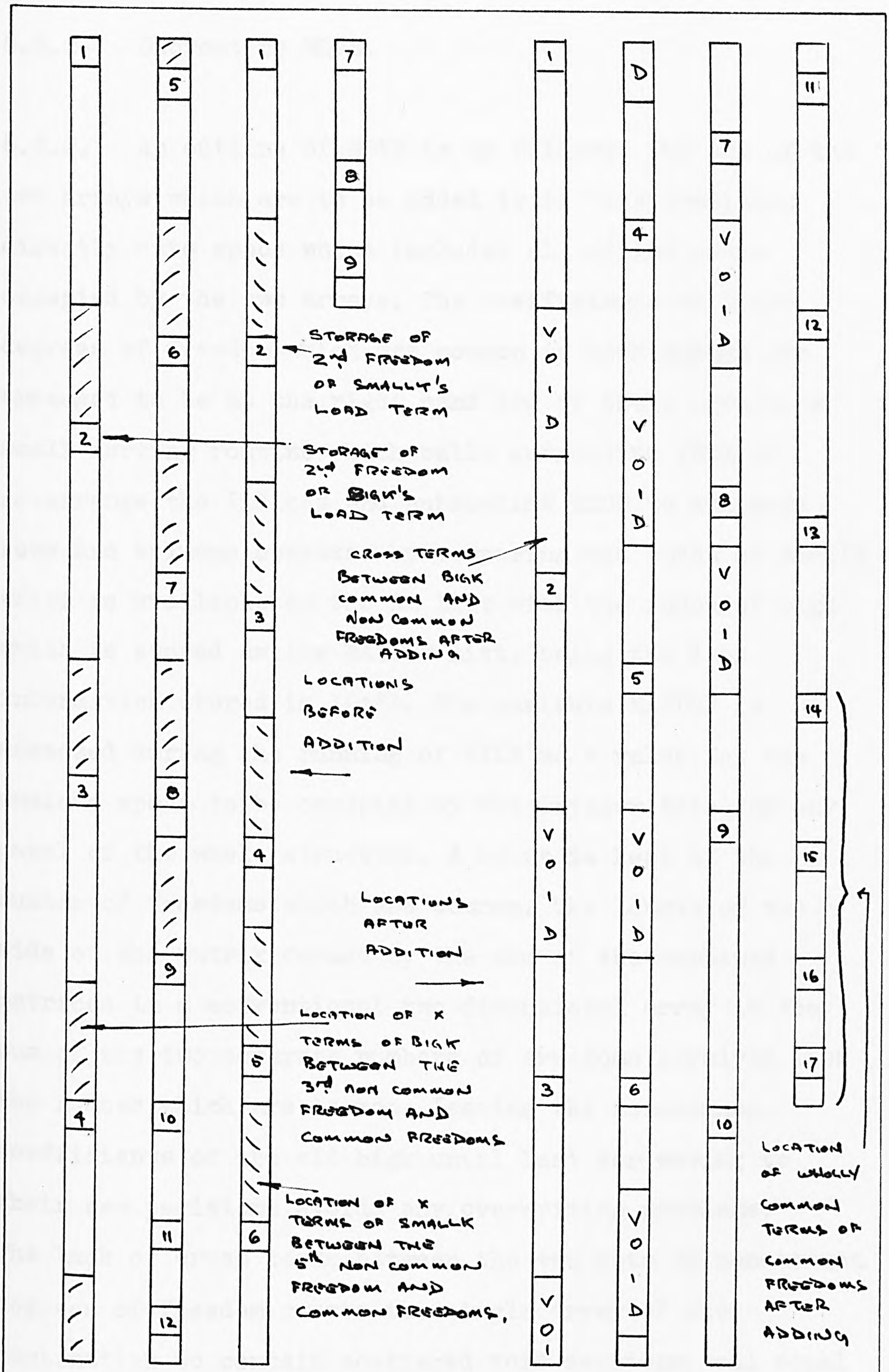
8.5.1 Let the stiffness matrix of that part of the structure assembled before the current panel was considered be called bigk and the stiffness matrix of the current panel in its reduced form of being expressed only in terms of external freedoms be called smallk, as represented in Fig. 8.8.

8.5.1.1 The addition of smallk and bigk and the arrangements of the indices is undertaken by MOVE.



Two dimensional representation of
the single dimension stiffness matrices.

Fig. 8.8



Actual locations of storages before and after addition of arrays shown in Fig. 8.8

Fig. 8.9

8.5.2 Subroutine MOVE.

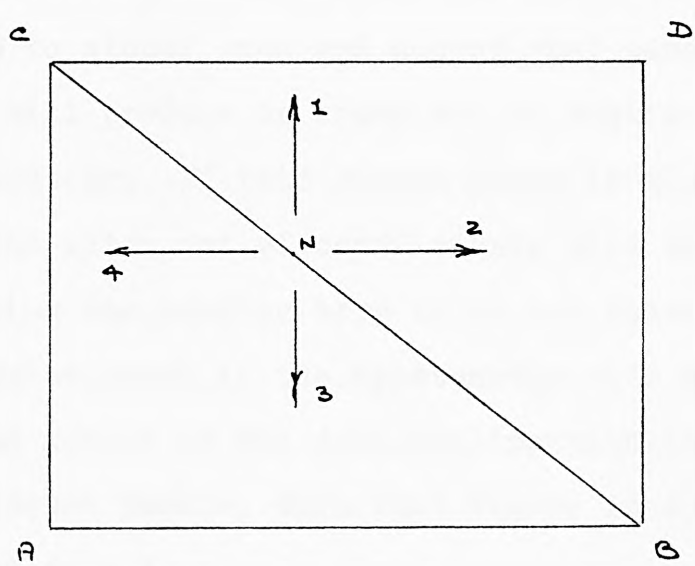
8.5.2.1 An outline of MOVE is as follows. The sum of the two arrays which are to be added is to be overwritten directly onto space which includes all of the space occupied by the two arrays. The coefficients of these degrees of freedom which are common to both arrays are arranged to be at the right hand end of their arrays. A small sorting routine which calls subroutine INDX to re-arrange the indices and subroutine EXCH to exchange rows and columns operates by comparing the index of smallk which is available as the KE list with the index of bigk which is stored as the MAXDOF list, being the last information stored in IDATA. The variable MAXDOF is assessed during the running of FILE as a value for the maximum space to be occupied by the integer data for any panel of the whole structure. A count is kept of the number of freedoms which are common. The length of the side of the matrix formed by the sum of the combined matrices in a conventional two dimensional array is the sum of the two separate numbers of freedoms involved less the number which are common. Leaving the non-common coefficients of the old bigk until last for moving to their new positions avoids any overwriting problems.
The lack of cross terms between the two sets of non-common degrees of freedom causes the single array of the combination to contain scattered void sections, all equal in length to the number of freedoms in the smallk, as represented in Fig. 8.9. It is not always so, that the sum of the two matrices will occupy a greater space than the

two hold separately, in which case, to avoid overwriting problems, the right hand end of the combined array must be moved to the right hand end of smallk. In such a case after adding, the array is moved to the left to vacate space for the next smallk.

8.5.3 As a result of the two arrays being added one or more edges may have left the wavefront and so the MAXDOF list indicating degrees of freedom which may be marked as inactive must be updated. To check whether a side has left the wavefront a scan is made of the corner activity of the current panel. When a zero is found it is known that degrees of freedom on sides having this corner as one end must all be available for consideration as inactive. It is the MAXDOF list of bigk which is to be modified but it is the KL list of smallk which must be searched. The presence of a spring ending on one of these listed freedoms will keep it active and a search must be made through the list of spring endings for such an occurance to prevent it being made negative. This list is stored immediately before the start of the MAXDOF list. Spring ends on internal freedoms initially cause these freedoms to be treated as internal restraints and as such are corrected back to positive during REACT. MOVE only scans the KL list for values to remove. This is satisfactory because MAXDOF list is formed from the KE list of the current panel and the MAXDOF list of the structure so far assembled. Therefore there is only the KL list to scan for the remainder. It remains only to see how the space occupied by the new bigk can be reduced to give more space to the arrays yet to be processed.

8.5.3.1 The matrix mathematics of the manipulation has been shown already in this chapter. A scan is made of the MAXDOF list from the left hand end and any negative values found trigger the process of exchanging the value's corresponding rows and columns with the most extreme right hand non negative value. The index value must also be changed.

8.5.4 If the current panel is a triangle, an out-of-plane degree of freedom reference number will have been introduced, even if there is no out-of-plane term. Therefore at this stage a scan is made of the leading diagonal terms and any non existant freedoms suppressed.



Incompatability of degrees of freedom along a common hypoteneuse.

Fig. 8.10

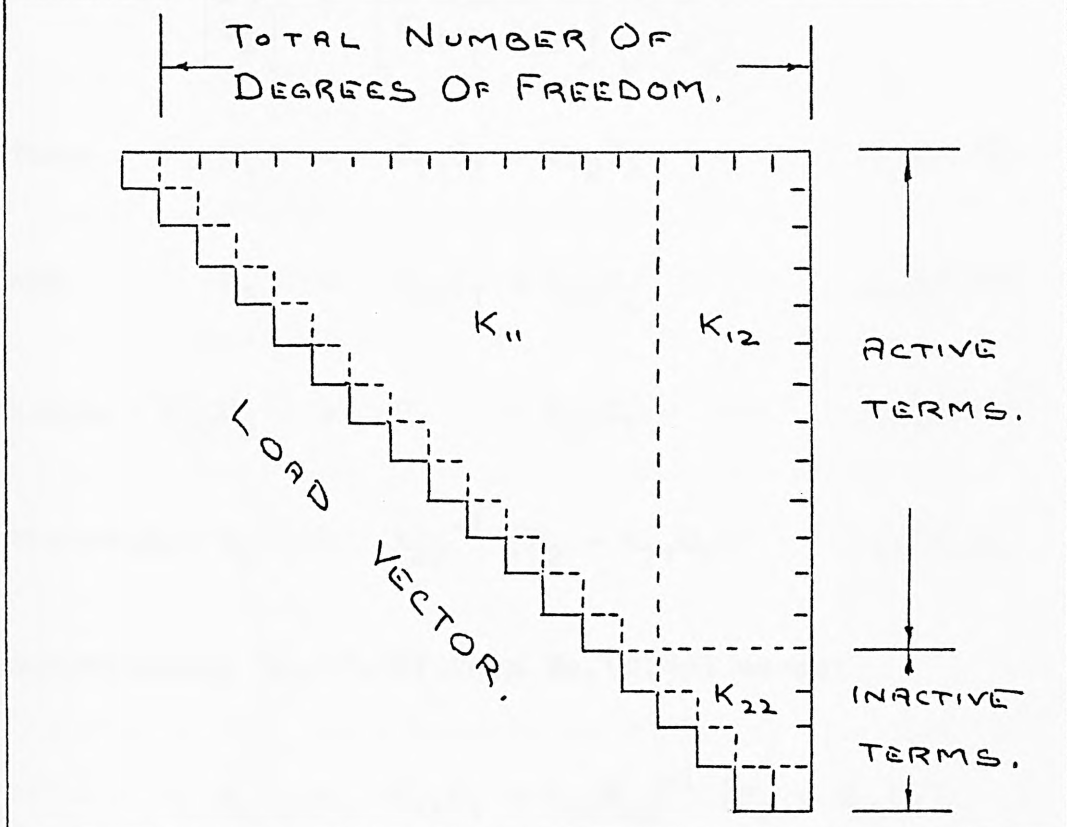
8.5.4.1 The reason for the introduction of an apparently unnecessary degree of freedom can be seen by considering two triangular panels ABC and BCD joining to form the rectangle ABCD as in Fig. 8.10. By the convention chosen for dealing with triangular panels, the local x and y axes of ABC are AB and AC so that the degrees of freedom at node N are those shown as 1 and 2. Similarly for BCD, where D is the origin they are 3 and 4. Obviously these are not compatible for direct addition and a transformation is required. One way to deal with this would be to derive the direction cosines of one panel relative to the other, a transformation in this case would not introduce an extra degree of freedom. With the exception of incompatible rotational freedoms, the remainder of the program is designed to transform freedoms when necessary to global reference axes. To keep consistency in transforming, it was decided to transform both sets of equations for the node N to global axes and accept that panels in the same plane will produce an extra set of coefficients that is not necessary. If this common plane is also a global plane then the extra set of coefficients will all be zeros requiring the leading term to be put equal to 1. For a triangular panel if the hypoteneuse side has had a 2 entered for it in the data dealing with the relative slopes of adjacent panels, then that figure is automatically changed to a 1.

8.5.4.2 An alternative could be introduced as suggested in Ref. (43) for the automatic constraining of linear dependencies. It was felt that this was an unnecessary addition to the program as that work by Kark and Scholes is designed for packages in which the analyst is responsible for identifying and constraining spurious linear dependencies and in the work described in these chapters, because of the assembly techniques developed, only in the few cases of triangular panels in the same plane will any unwanted freedoms appear and these can be easily dealt with as above.

8.6 Compressing the matrix after addition of the most recent panel.

8.6.1 As a result of the action of TRANS, the current stiffness matrix may now be added to the stiffness matrix of that part of the structure so far assembled. After the two matrices have been combined into a single array, the index of the combination is scanned to see if there are any degrees of freedom present which are no longer on the wavefront. If there are any such freedoms, then the new array is re-expressed in terms of only its own freedoms which are still active plus freedoms of spring ends being retained until the final inversion stage. Unlike the position which exists at the initial assembly of each local stiffness matrix, where it is successfully prevented from ever occupying its full size, in this the stiffness matrix is fully developed for this point.

8.6.1.1 Since no more coefficients are going to be added until the next panel's stiffness matrix is formed, the use of a technique whereby a complete matrix is expressed in a smaller number of terms may be undertaken.



Partitioning of Bigk.

Fig. 8.11

This technique involves the partitioning of rows and columns of the direct and cross terms of inactive freedoms and by a substitution process, the modification of the remaining matrix and load vector for the effect of their removal.

8.6.1.2 The matrix operations for the technique are as follows.

Let the relationship $P = \left[K\right] U$ of an elastic system be partitioned into parts representing the direct terms to be retained and those terms which are not to be referred to in the final form.

So that $\left[\begin{array}{c} P_1 \\ \hline P_2 \end{array}\right] = \left[\begin{array}{c|c} K_{11} & K_{12} \\ \hline K_{21} & K_{22} \end{array}\right] \left[\begin{array}{c} U_1 \\ \hline U_2 \end{array}\right]$

then $\quad P_1 = K_{11}U_1 + K_{12}U_2 \qquad \ldots(8.15)$

and $\quad P_2 = K_{21}U_1 + K_{22}U_2 \qquad \ldots(8.16)$

hence $\quad K_{22}U_2 = P_2 - K_{21}U_1 \qquad \ldots(8.17)$

therefore $\quad U_2 = K_{22}^{-1}(P_2 - K_{21}U_1) \qquad \ldots(8.18)$

Substituting Eq.(8.18) into Eq.(8.15) we get

$$\begin{aligned} P_1 &= K_{11}U_1 + K_{12}K_{22}^{-1}(P_2 - K_{21}U_1) \\ &= K_{11}U_1 + K_{12}K_{22}^{-1} - K_{12}K_{22}^{-1}K_{21}U_1 \end{aligned}$$

hence $\quad P_1 - K_{12}K_{22}^{-1}P_2 = (K_{11} - K_{12}K_{22}^{-1}K_{21})\, U_1$

which is of the form

$$P_{mod} = K_{mod}U.$$

We may therefore solve for displacements of the upper partitioned zone solving a reduced stiffness matrix size.

Hence $\quad U = K_{mod}^{-1}\, P_{mod}.$

Substitute back into Eq.(8.16) and the solution of the remainder of the displacements may be found.

i.e. $U_2 = K_{22}^{-1}P_2 - K_{22}^{-1}K_{21}U_1$.

8.6.2 Compressing the reduced matrix assembly.

8.6.2.1 The bigk matrix is partitioned as shown in Fig. (8.11) and after processing, the full inverse of K_{22} is stored so that the space used for the half of K_{22} is overwritten. The load vector associated with the inactive terms is moved to the right to enable it not to be overwritten and to be in a suitable space for copying to disc.

8.6.2.2 K_{12} is transferred to a position right behind the inverse of K_{22}, as advantage is taken of the transfer to reform it as K_{21}. This block consisting of K_{22}^{-1}, K_{21}, and the inactive load vector are now all put off to disc as a single vector. Reference data such as the current global index and previous disc record number are also put off to disc.

8.6.2.3 The product $K_{12} * K_{22}^{-1}$ is formed and the result stored in the space previously used for K_{21} which has been put off to disc. Since this product has the same size as K_{21}, the inactive load vector is not disturbed. The operation to modify the active load vector can proceed, followed by the modification of K_{11}. A last operation is to move the scattered parts of K_{11} and the load vector to the left, overwriting the now not required K_{12}.

9. Inversion and iteration.

9.1 Modifying the stiffness matrix to allow for restraints being transmitted to the structure via springs.

9.1.1 It was explained in section 2.6 that there are no fixed degrees of freedom with a freely floating body. To avoid a singular stiffness matrix in such a case it was decided as outlined in Ref. (44) to anchor the structure to hypothetical rigid points by sufficient springs to counter all rigid body motions.

9.1.1.1 It is necessary so that the final matrix does not become ill-conditioned that these springs be stiff in comparison with the structure.

9.1.1.2 To achieve this the addition of the stiffness terms of these springs is left until after the stiffness matrix of the remainder of the structure is assembled. A loop is then used to scan the leading diagonal terms of that matrix to find the largest value. These anchoring springs are then given a stiffness value of ten times the maximum value found.

9.1.2 A second type of spring providing restraint is available, which permits a part of a structure to be analysed. This type of spring represents the support given to the structure by the adjacent structure which cannot be allowed for by symmetry and is unable to be added to the stiffness matrix conventionally. In this case the stiffness value of the individual springs must be estimated by the analyst and entered in the data.

9.2 Modifying the stiffness matrix to allow for the insulating blocks.

9.2.1 In the type of problem which this package was designed to solve, there can be considered two principal structures. One is the cargo tanks maintained at a low temperature. The other is the hull of the ship built mostly of mild steel and not suitable for the temperature at which the cargo is kept. Insulating material is used to lessen the heat flow between the two. The cargo tanks are insulated to reduce heat transference through the separation space and where as in the case of load bearing supports, physical contact is necessary, insulation blocks of wood or composite material are used. To allow for relative expansion and contraction of the two structures the tanks are not secured to the blocks by mechanical means. Since the tanks and the hull are not fastened to each other through the supporting blocks no tensile load can be passed between them. The only type of loading which can be considered to be applied to the blocks is compression. In this text these blocks will usually be referrred to as springs. It will usually be necessary to idealize each block by several springs if the load is not uniform.

9.3 Handling spring contribution to the stiffness matrix.

9.3.1 It will be seen from input details given in Appendix A that the last item for each panel is a set of integers, being the number of the node at which the spring is attached to that panel, the order of the freedom at that node in which the spring acts, the hand allocated spring number and the type of spring being applied. The line of action of the spring must be coincident with that of the declared freedom, this requirement invalidates the use of any rotational degree of freedom to be at either end of an insulating block type spring. In the manner used for local loads and restraints there is a built in routine to convert the local numbering system of node number followed by the freedom number at that node, first to a local freedom number counting from the local origin and subsequently to global numbering. There is no restriction on whether the spring end aligns with an external or internal freedom. During ASSEMB if the spring end is internal, like a restraint it is treated as a pseudo external freedom and remains in the local stiffness matrix. Unlike an internal restraint though, during COMPRS the freedoms of spring endings are not eligible to be considered as inactive even when there is no other freedom of that global number left to be added by another panel.

9.3.2 At the completion of COMPRS for the last panel of the whole assembly, the stiffness matrix consists of only those degrees of freedom which are also the degrees of freedom of springs plus some freedoms from the last panel which were not on a common side and were therefore not on the initial list of freedoms which had become inactive. Had this not been the last panel, at the second check for inactives done by looking for inactive corners, these freedoms would have been found, but this second search is not considered necessary for the final panel, since the space that would have been provided is not now at a premium.

9.4 Completing the final matrix.

9.4.1 The adding of the springs representing the insulating blocks may be thought of as a dynamic addition since they are subject to removal from the apparent final matrix to form a revised final matrix. Before this process is started it is necessary to add any anchor springs and springs representing adjacent structure which are going to stay in the matrix.

9.4.2 Identification by the program of which type of spring is being currently considered is done by scanning the spring data. At the initial scan of this data, which at this stage is only integer values, each spring's data relating to its possible fixed end is examined. If this location contains a freedom number then that spring must be representing an insulating block. This can be deduced as other spring types do not have a freedom number allocated for their restrained end.

9.4.2.1 If the spring is an insulating block then that spring is ignored at this stage and the next spring is examined. If this is not found to represent an insulating block then a further examination is made of its data. Anchor springs as already explained have their stiffness value automatically calculated from the stiffness matrix and therefore there is no need for the analyst to nominate a type for that spring other than a zero integer. This second data search then looks at the type number, if the type is zero, the spring's leading diagonal term is the calculated stiffness value. A positive type number triggers a search to be made through the supplied real number spring data to find the allocated stiffness value.

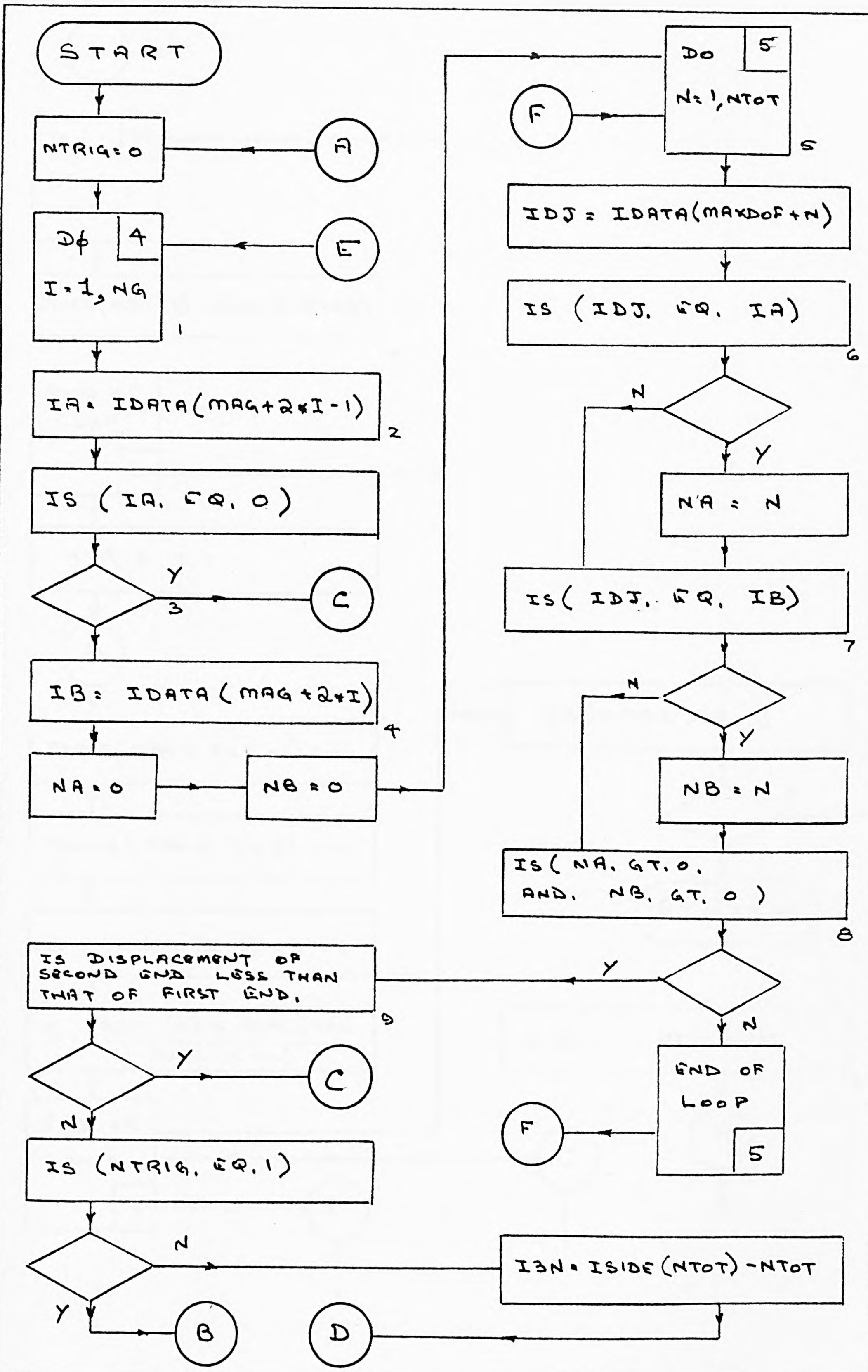
9.4.2.2 However the stiffness value is obtained, it is put into the leading diagonal position indicated by the freedom number of the free end of the spring. Since the other end of the spring is restrained there is no need to add any other term.

9.4.3 There is no need to increase the total number of degrees of freedom in the structure to allow for anchor or adjacent structure representing spring ends because the end of the spring attached to the structure has a freedom number which was allocated to the structure at the start of ASSEMB. At the other end of the spring it is totally restrained therefore the matrix for inversion need not be expanded as this area of the matrix would not affect the inversion and hence there is no need to keep an index of the restrained ends of springs. Similarly it is not necessary to add any space to the final stiffness matrix for the addition of rows and columns due to the fixed ends of springs. Before moving on to examine the next spring's data, the reference number of its free end's degree of freedom is put to zero to avoid complications caused by incorrectly initiated searches during the eventual iteration loop.

9.4.4 When the panel which was seen immediately before the current panel was assembled into the reduced stiffness matrix of the remainder, the space occupied in the core could not be greater than the value of the variable LIMIT and when this combined stiffness matrix was processed by COMPRS there must have been space created in proportion to all those freedoms then considered inactive. This space is now available as an overflow store for the details of the springs. This is only necessary if they prove too numerous for the primary storage space. There is no use at this stage for the space on the right hand side of LIMIT which has until now been used for the storage of the formulae of the element stiffness matrices. Spring details may overwrite this area and use it as a primary storage area. If there is not sufficient space to store the data for each spring to the right of LIMIT then the value of LIMIT is revised to give exactly the space required by including sufficient secondary space. The data to be read in for the spring information consists of the block's or spring's stiffness value, being the load to compress it a unit distance. In the case of blocks account must be taken of the number of springs being used to represent a particular block. To save repeating the input data each value is prefixed with an integer being the number of successive springs having this value. Once this data is read in, the inner of a pair of nested loops allocates this temporary data to its permanent location and the outer loop controls the number of times this block of data is repeated.

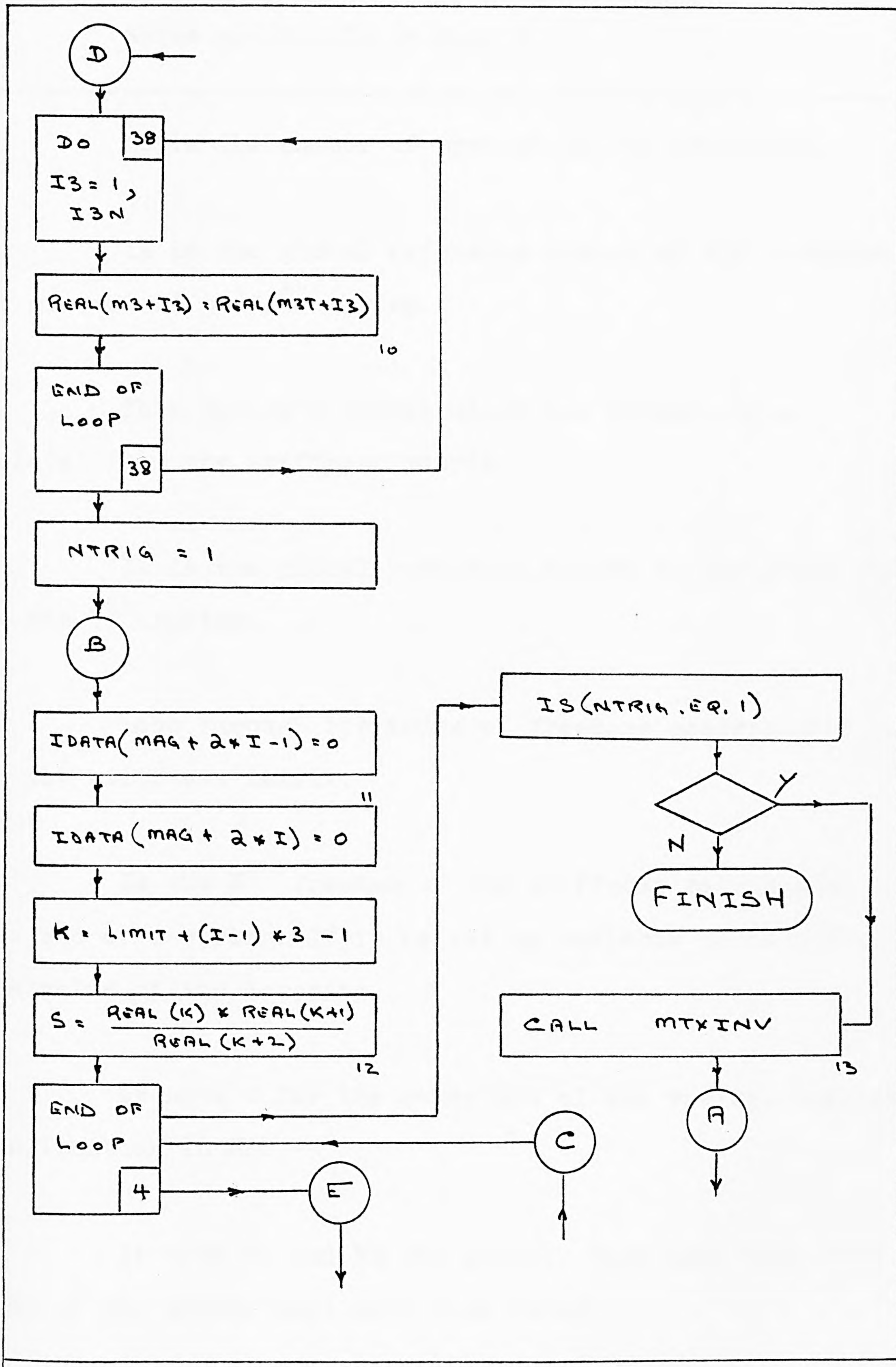
9.4.5 During ASSEMB a list was built up giving the global reference numbers of the freedoms at the ends of each spring. This was achieved by examining for each panel the input data relating to springs which was mentioned in 9.3.1 consisting of two integers per spring. After the spring number is read the global reference number of the spring end is found and copied into one of a pair of locations which in total form the MAG list. The spring number controls the distance into the MAG list for the spring's pair of locations. If when attempting to find the location for the reference number of the freedom of the spring end the storage space is found to be already occupied then the value is put in the second location of the pair. The first being filled indicates that the second end of the spring is now being examined and since there is an active second end this must be an insulating block.

9.4.6 To complete the final stiffness matrix, the stiffness terms for the supporting blocks must be added. A pair of nested loops is set up to add the stiffness terms of the remaining springs to the overall stiffness matrix. The outer loop controls the processing of one spring at a time and does not move to the next until the



Flow chart for the removal from the final stiffness matrix of the stiffness terms of those terms of springs in tension. Part 1

Fig. 9.1(a)



Flow chart for the removal from the final stiffness matrix of the tiffness terms of those terms of springs in tension. Part 2

Fig. 9.1(b)

Notes applicable to Fig. 9.1

1. NG is the number of springs in the structure.

2. IA is the global reference number of the freedoms at one end of the I^{th} spring.

3. This spring's contribution has already been deleted from the stiffness matrix.

4. IB is the global reference number of the other end of the I^{th} spring.

5. Loop through the index of freedoms contributing to the stiffness matrix.

6. Is the N^{th} freedom of the stiffness matrix also the end of a spring. If it is set up variable NA to store the value of its location.

7. As note 6 for the other end of the spring, storing its location in NB.

8. If both NA and NB are greater than zero then both ends of the spring must have been found.

9. Compare displacements of each end of spring and make a decision whether the spring is in tension or not.

10. If not already recovered, recover the stored

pre-inversion version of stiffness matrix. Variable NTRIG serves to indicate if matrix has been recovered.

11. Delete from list of springs, the spring which has been found to be in tension.

12. As when assembling spring terms now use same statements to remove terms.

13. Since the effects of one or more springs have been removed from the matrix, invert the revised matrix.

other is fully assembled into the stiffness matrix. The outer loop steps through the MAG list selecting the global reference numbers of the degrees of freedom of the two ends of each spring and giving them the variable names IA and IB. At this time two other variables NA and NB are set to zero. Control is now passed to the inner loop which proceeds to search the NTOT positions of the MAXDOF list, being the index of the final stiffness matrix where NTOT is the number of freedoms in the final array, until the locations in the matrix of the spring's ends are found. The current value of the MAXDOF list is given the variable name IDJ. If IDJ equals IA the variable NA is given the value of the inner loop counter. Similarly if IDJ equals IB, NB takes the counter's value. When both NA and NB are not set to zero control can jump out of the inner loop and back to the outer loop since the locations in the matrix of both ends of the spring are known. Before the outer loop moves on to the next spring it must recover details of the current spring and load them into the stiffness matrix. 9.4.4 explains details regarding the properties of springs, such details are stored in numerical order in the array Real after the location LIMIT. The outer loop counter will give the key to its location.

9.4.6.1 The location for the leading diagonal space for the stiffness matrix is found by putting the variables NA and NB into the function IPART. The location of the cross product term uses both NA and NB but since there is no requirement for either to be specified as the larger, the composite Fortran function statement

IPART(NTOT,MINO(NA,NB)) + 2 + IABS(NB-NA)

is used to find its location from the start of the stiffness matrix. There can only be one value going into the cross product location. It is therefore arranged that if the location is not void before adding the term, the run will stop and a fault message be printed.

9.5 Inverse of the final stiffness matrix.

9.5.1 A number of equation solving algorithms are available and a review of some of these based on Gauss elimination was given by Mondkar and Powell (45). A wider review has been undertaken by Meyer (46) which contains 146 references to problems related to the solution of linear equations. Most of these approaches have been devised with sparce matrices in mind. The reduction and compression techniques used in this work results in a dense final matrix for inversion.

9.5.2 Fox and Meyer (47) give the following brief mention of inversion: "Even if we have the same matrix with a number of different right hand sides it is faster to perform the triangular decomposition, once and for all followed by forward and backward substitution for each right hand side. We may need the inverse in its own right. Its computation is performed one column at a time, by solving linear equations with the same matrix on the left and successive columns of the unit matrix on the right".

9.5.2.1 Using this note as guidance and taking advantage of the fact that the matrix for inversion is symmetrical the subroutine K22INV was developed, based on the Choleski decomposition technique.

9.5.3 The following shows the derivation of the features of K22INV.

Let $P = [K]X$ represent a set of equations, where X is the unknown vector to be found.

Substitute $[U^T][D][U]$ for $[K]$

where $[U]$ is an upper triangular matrix with

$$U_{rs} = K_{rs} - \sum_{i=1}^{i=r-1} \frac{u_{ir} * u_{is}}{u_{ii}} \qquad \text{... (9.2)}$$

and D is a diagonal matrix

with $D_{ii} = 1/U_{ii}$

hence $P = [U^T][D][U]X$

putting $Y = [D][U]X$ ... (9.3)

gives $P = [U^T]Y$

Since $[U^T]$ is a triangular matrix and the column P is known, the column Y may be derived by calculating directly the last Y value by dividing the related P value by the only term of the $[U^T]$ matrix that is in that row. Subsequent Y values may be derived by a similar operation but in these cases the previously found Y values must be substituted.

9.5.3.1 The outline of the routine is to form the $[U]$ triangular matrix once and then obtain and store new X columns using the above method for each P column in turn where these are the columns taken in turn from an identity matrix.

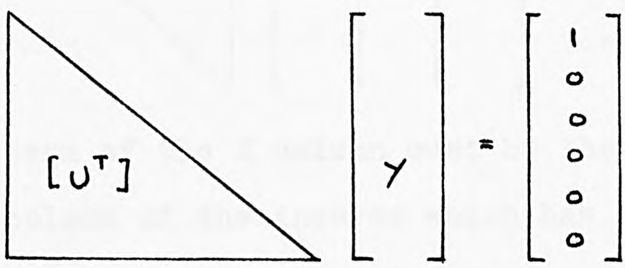
9.5.3.2 The array [U] is actually a vector formed of a segment of the array Real, immediately to the right of the structure's stiffness matrix storage. The current column of the identity matrix is stored to the right of the [U] matrix and is transferred to the current column of the matrix inverse. When the column is complete it is transferred back and overwrites the appropriate space in the core storage of the matrix being inverted. The column to the right of the [U] matrix is emptied and the digit one entered in the i^{th} location, where i is the loop counter value. The inverse is also symmetrical therefore each new column of the inverse may be calculated for one row less than the last. It can be seen from Eq. (9.2) that each row of the matrix [U] is a function of the terms above it.

9.5.3.3 The process starts as shown, with the relationship $P = [U^T]Y$, where P is the first column of the identity matrix and Y is related to X, the first column of the matrix inverse by Eq. (9.3).

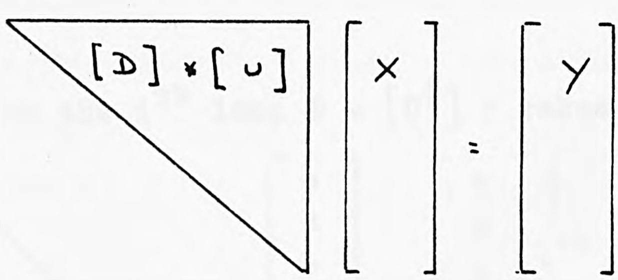
9.5.3.4 To obtain the subsequent columns of the inverse the above process may be repeated with the appropriate column of the identity matrix being used for P. This is, however, wasteful since the inverse is symmetrical, for the i^{th} column the first (i-1) values must already exist and therefore it would seem that the process could be stopped a row earlier after each loop. But there is not a direct relationship between [U], P and X. P is directly related to Y and Y is related through [D] and [U], to X. It is not axiomatic that the operation could be stopped a row earlier after the formation of each column of the inverse. Further, if we are now to operate on a reduced size of $[U^T]$ it may be that a fresh [U] should be obtained since if the upper part of the full $[U^T]$ is not considered the lower terms do not now hold correctly defined relationships to the terms above them.

9.5.3.5 The following shows that the disposition of zeros at the top of the current column of the identity matrix allows the comments of the last paragraph to be ignored.

At the first loop we have for $P = [U^T]\,Y$

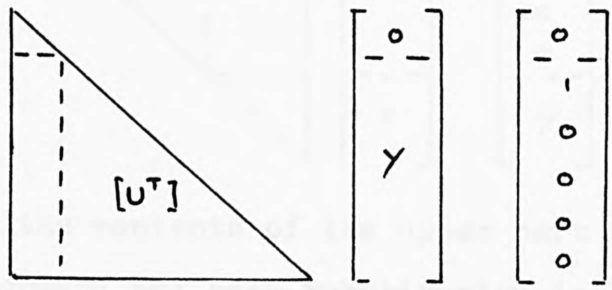


hence the Y column is obtained and may be substituted into $Y = [D] * [U]\,X$ to give

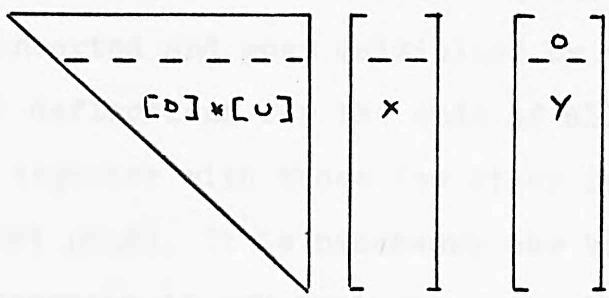


hence the column X is obtained and put into K_{22} by overwriting the appropriate spaces to start forming the inverse of K_{22}.

At the second loop the P column is revised and we obtain for $P = [U^T]Y$,

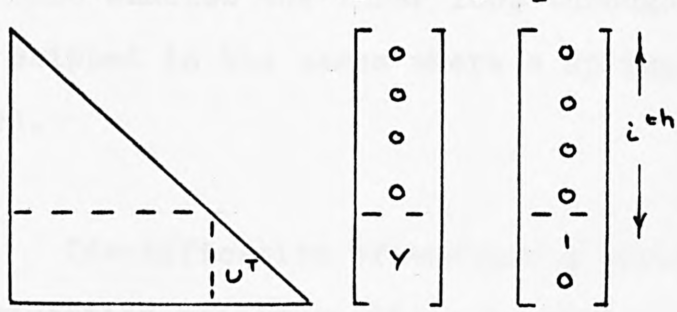


Since the first term in P is a zero, the first term in Y must be a zero. Hence the first column of $[U^T]$ may be retained unchanged but ignored. The Y column thus obtained may be substituted into $Y = [D] * [U]\ X$ to give

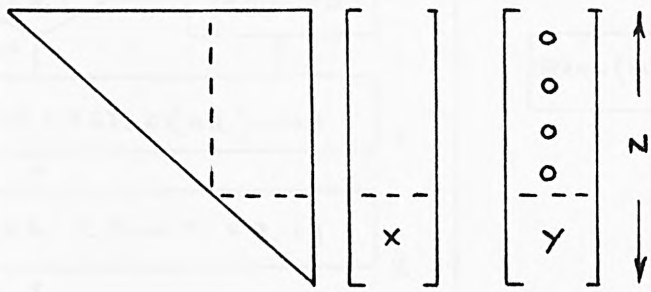


The first term of the X column must be the second term of the first column of the inverse which has already been found. Since we are now substituting from the bottom upwards there is no need to consider the top row of $[D] * [U]$.

For the i^{th} loop $P = [U^T]\ Y$ takes on the appearence



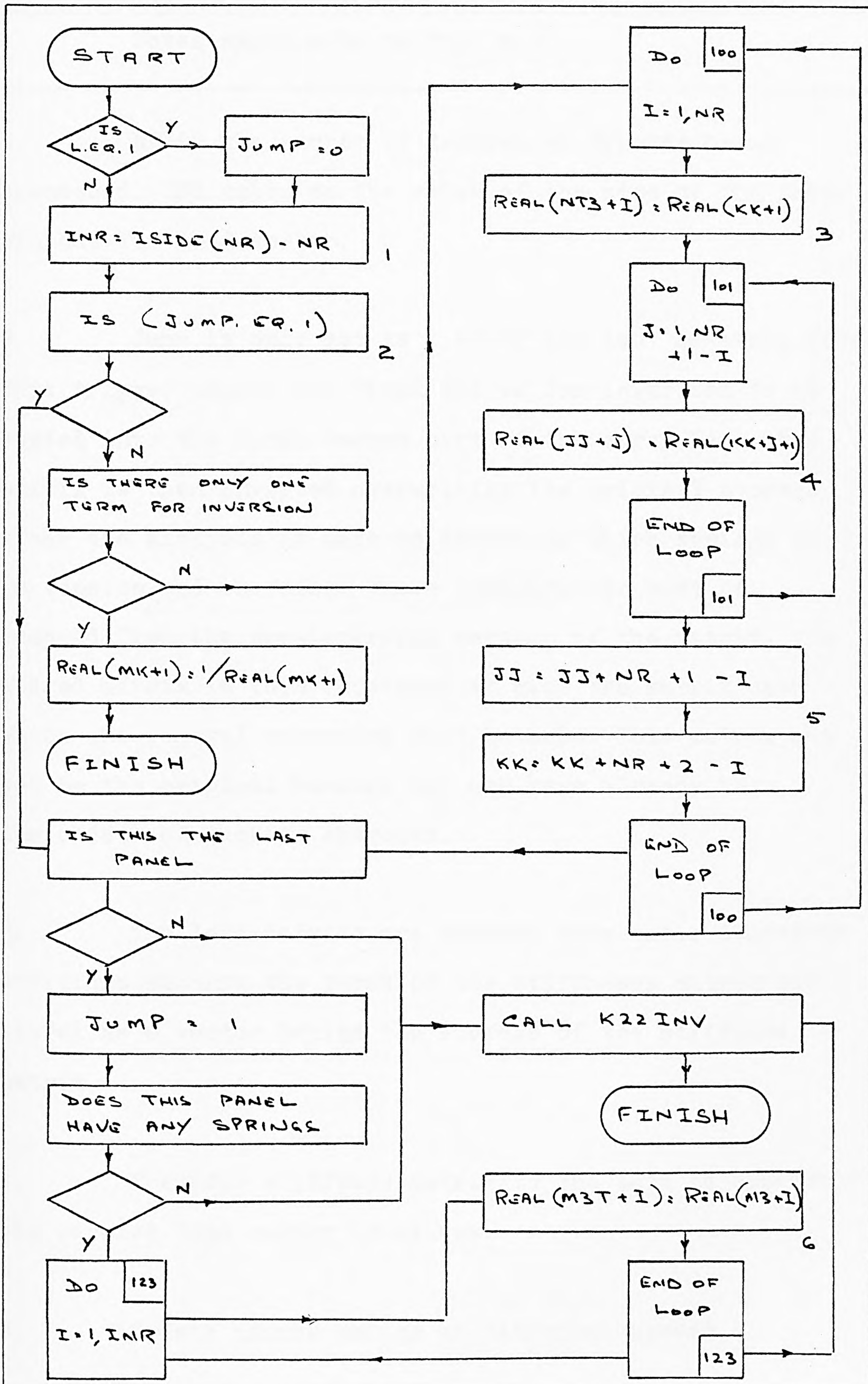
The relationship $Y = [D] * [U] X$ has for the i^{th} loop the appearance,



where the contents of the upper part of X are already known. The forward and back substitution is limited on the i^{th} loop to $N - i + 1$ rows.

9.6 The final stiffness matrix is now fully assembled and is inverted in the way indicated. When the matrix has been inverted and post multiplied by the load vector, a set of deflections for the ends of all the springs will be found together with those few other freedoms remaining from the last panel. It is necessary now to examine these results and ascertain if any springs are in tension and if they are, to delete their contribution to the overall stiffness matrix. The search for such springs is conducted in a manner similar to the technique given in 9.4.6 for finding where coefficients due to springs should go in the overall matrix. Additionally if any spring is found to be in tension its two reference numbers are deleted from the MAG list. This enables the inner loop through the MAXDOF list to be skipped in the cases where a spring has already been deleted.

9.6.1 Identification of whether a spring is in tension or compression relies on the automatic generation of



Flow chart showing control of inversion routine.

Fig. 9.2

Notes applicable to Fig. 9.2

1. NR is the number of degrees of freedom being processed. INR takes on the value of the size of the array without the load vector.

2. Jump is only set to 1 after the las+ assembly loop. This trigger causes the final matrix for inversion to be copied into the first vacant part of the array Real. The matrix is then inverted overwriting its original storage. After the analysis is made to determine which springs are in tension and therefore whose contribution must be removed from the pre-inversion version of the matrix, the stored matrix is then recovered to give the matrix upon which the removal operation must be made. This matrix may not be the original version but may have already been the object of such an exercise.

3. The load details are removed from their scattered positions amongst the terms of the stiffnesss matrix and stored as a vector behind the storage of the stiffness matrix.

4. Transfer stiffness matrix to the left to overwrite the vacated load vector locations.

5. Update counts moving at different speeds.

6. Store matrix as mentioned in note 2 above.

spring end global reference numbers being always smaller on the first of the two panels concerned to give a direction sense to the line of the spring. It is necessary when allocating by hand a freedom to be the end of a spring, that it acts in the line of action of the spring and for a positive deflection at the time of inversion in the same direction as the other end, which is not necessarily the local positive direction. For a spring a positive deflection is a deflection in the direction from the first mentioned end towards the second.

9.6.2 When a spring is identified as being in tension its stiffness value is found as before and deleted from the final stiffness matrix in an exactly opposite way to its original assembly. A reference that this spring is no longer used in the analysis is printed out.

9.6.3 A trigger is set from zero to one if there is one or more spring terms to remove. This serves to return control of the run back to the inversion routine. This iteration process of looping back to invert the matrix again without the terms of springs which were found to be in tension, is repeated until all the remaining springs are in compression. At the last loop the trigger remains unprimed at zero because there was no presence of springs in tension to set it to one. A final examination is made of the MAG list to find the reference numbers of the remaining springs. Using these numbers the net deflections of the springs are found and their loads evaluated and printed out.

9.7 The final stiffness matrix is not the only matrix needing to be inverted. During COMPRS the operation to express coefficients of the inactive degrees of freedom in terms of these still active by using

$$P_a - K_{12} * K_{22}^{-1} * P_i = (K_{11} - K_{12} * K_{22}^{-1} * K_{21}) U_a$$

necessitated the inversion of part of a matrix.

9.7.1 The same subroutine K22INV is used for inverting both K_{22} and the whole of the final matrix. In the case of K_{22} it is not required any more and may be overwritten to save space for K_{22}^{-1}. Since the matrix of the final stiffness may be needed again if it contains coefficients of springs which later prove to be in tension. Therefore the final stiffness matrix must not be overwritten by its inverse until it is known that all the springs are in compression. To achieve these conflicting requirements K22INV is called by a controlling subroutine MTXINV which organizes the location of the matrix before and after inversion and moves theload vector to clear the inversion manipulations.

10 Displacement and stress evaluation.

10.1 Forming the displacement values of the external degrees of freedom of the current panel.

10.1.1 Consequent upon the inversion of the final stiffness matrix, a set of displacements will be found for those freedoms still active. That those may or may not be springs or even on the current panel is of no importance. The reconstruction of the panel idealization will be entirely controlled by ordered information stored on disc. The duty of EXPAND is to use those displacement values already known, to reform the relevant equations existing during the comparitive stage of COMPRS, to form a complete list of all the external displacements of the current panel. This list will have two uses, it will be the key to forming all the internal displacements and secondly it will supply the values needed to initiate this routine for the next panel. The reconstruction of panels proceeds in exactly the opposite order to that used before the final stiffness matrix inversion.

10.1.2 At the start of EXPAND whether or not this is the first panel after the inversion the index of active freedoms will be the Maxdof list. This list may contain freedoms which are not on the current panel but were active during COMPRS and are therefore necessary for reconstruction. If this is not the first panel after inversion the Maxdof list will contain the freedoms of the last panel which may or may not be necessary for the current panel's reconstruction. Information put off to disc during COMPRS contained amongst other things a list of all the active and inactive freedoms pertaining at that time and in the order they existed before the matrix operations of COMPRS were begun. The existing Maxdof list contains this information for the previous panel.

10.1.2.1 The core space occupied by this existing Maxdof list must now be cleared to make space to receive the current Maxdof list. But the existing Maxdof list contains information about the existing displacements, some of which are necessary to form the equations to find displacements of the current panel and possibly subsequent panels. The existing absolute values of the Maxdof list are therfore transferred to become the new KA list. For the next panel this newly arrived Maxdof list will in turn be moved to form the KA list.

10.1.3 At this time there exists a list of displacement values in a segment of the array Real of length equal to the variable NTT, the content of this segment is now indexed in the KA list. The new Maxdof list is examined and at each of its locations a scan is made of the KA list to find if the degree of freedom reference number occupying that location is present anywhere in the existing values. If on the i^{th} search of the KA list a reference number is found equal to the Maxdof list number which is the focus of the i^{th} search and if that number is the j^{th} value of the KA list, then the i^{th} and j^{th} values of the M3 list are exchanged. To avoid confusion to subsequent searches the i^{th} and j^{th} values of the KA list are also exchanged. When this stage is complete the list of displacement values of the M3 list will be in an order to agree with the list of active freedoms at the begining of the comparable COMPRS state. This order is given by the positive values of the new Maxdof list and has a length given by the variable NM. Values in array Real beyond the variable value NIM, where NIM = M3+NM are no longer required and had become inactive before this panel was reached by COMPRS. Their locations may now be overwritten.

10.1.3.1 External freedoms which were inactive during COMPRS for the panel being currently rebuilt are also on the Maxdof list but may be differentiated from the active freedoms by the presence of a negative prefix. It is the presence of these negative signs which necessitates the absolute values being transferred when forming a new KA list from the existing Maxdof list.

10.1.4 It may be remembered that during COMPRS the inverse of K_{22}, K_{21} and the inactive load vector were put off to disc. This was so that once the displacements of the degrees of freedom which were active during COMPRS have been found the inactive external and pseudo external displacements may be found by the matrix operation :

$$U_i = K_{22}^{-1} * P_i - K_{22}^{-1} \overset{*K_{21}}{\wedge} * U_a \qquad \ldots(10.1)$$

where U refers to the displacement vector

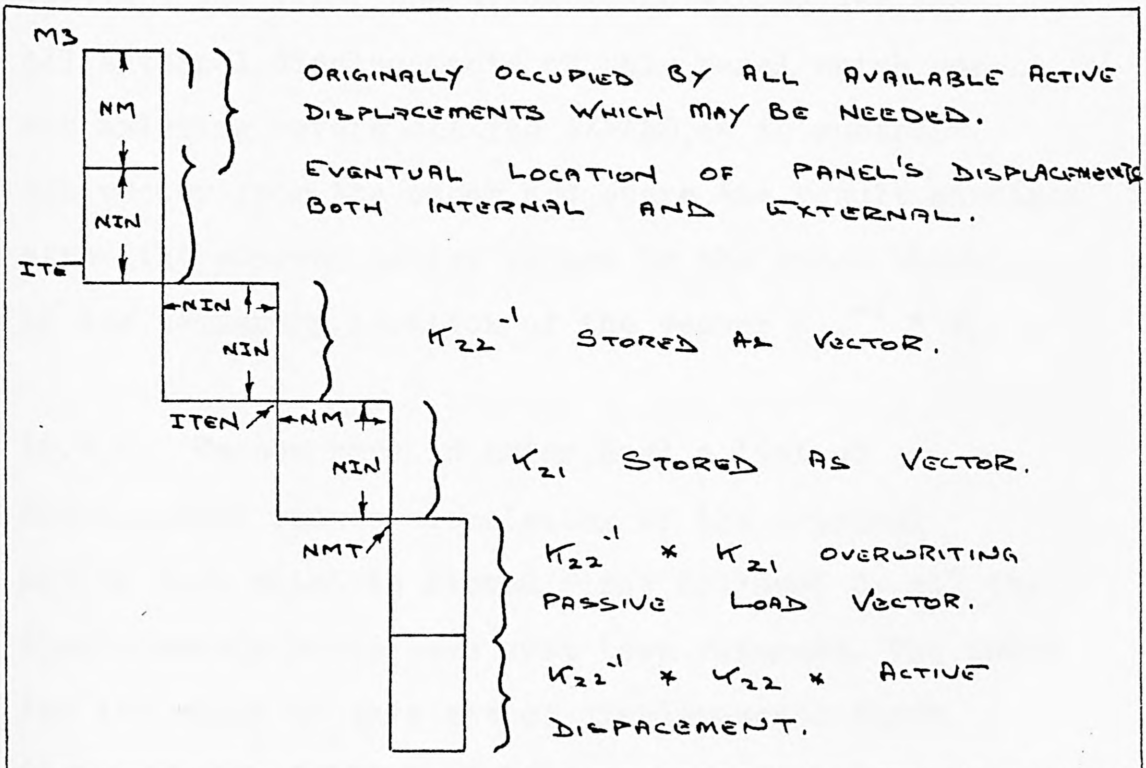
P refers to the load vector

i refers to an inactive mode

a refers to an active mode

K_{22}^{-1} is the inverse of the stiffness matrix of the inactive degrees of freedom, and

K_{21} is the matrix of the cross terms between the active and inactive terms.



STORAGE LOCATIONS OF K_{22}^{-1}, K_{21} and PASSIVE LOADS, and LOCATIONS FOR REFORMING MISSING DISPLACEMENTS.

FIG. 10.1

10.1.4.1 The operation $K_{22}^{-1} * P_i$ is performed first and the product is temporarily stored overwriting surplus displacement values left after re-arranging the available active displacements.

The operation $K_{22}^{-1} * K_{21}$ is performed one row at a time. This row is stored overwriting the inactive load vector space, the contents of this vector was used in the last operation and is not needed again. When the row is formed, the second stage of this operation, its post multiplication by the active displacement vector may be made and the single value resulting is stored in the next available space after the old inactive load vector storage.

10.1.4.2 The final operation to obtain values for all external displacements of this panel which were not existing before calling EXPAND, is to subtract one vector from the other and store the result straight after the current active values in the space which is the temparary location of the vector $K_{22}^{-1} * P_i$

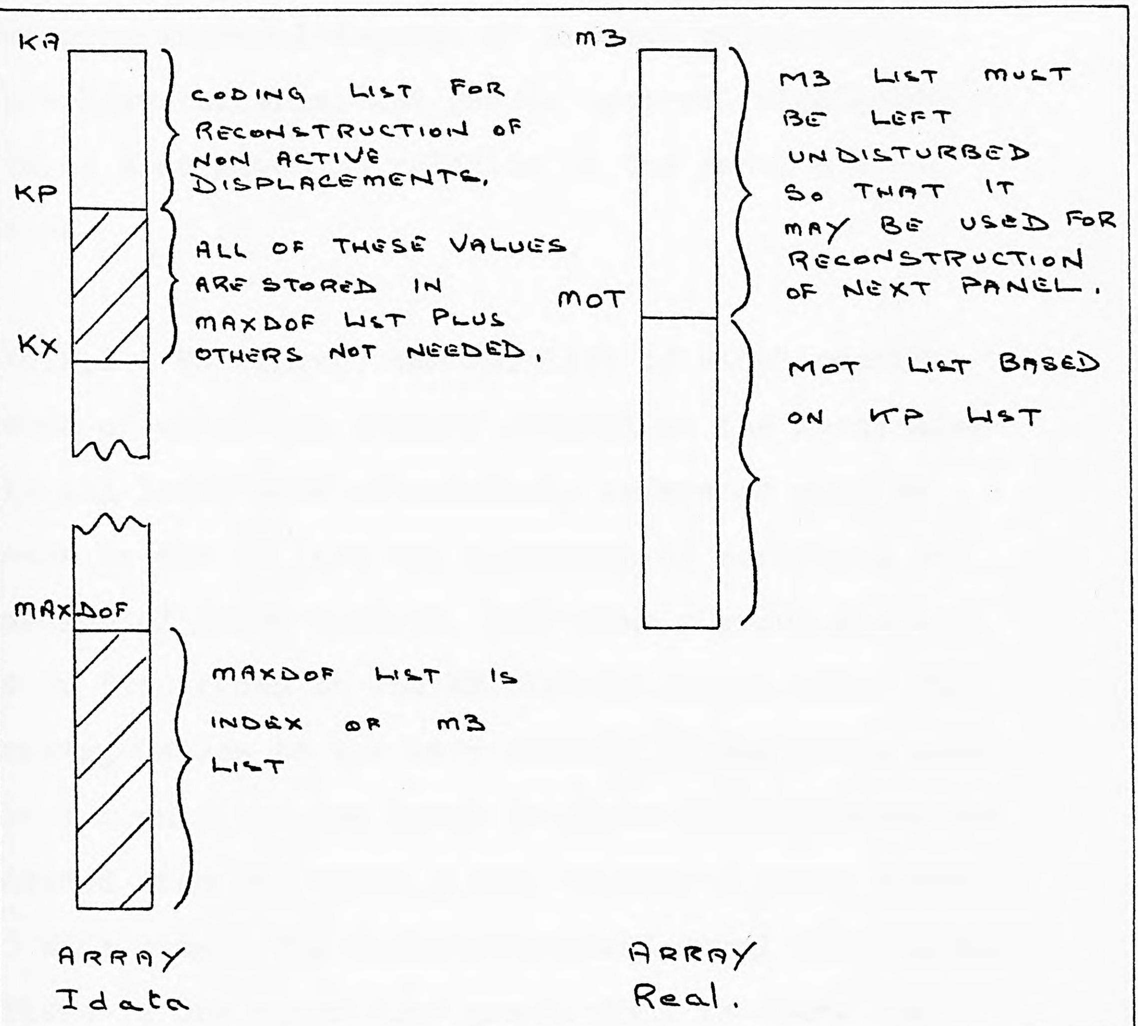
10.1.5 We now have in array Real a list of displacement values, consisting of the original active list which is stored first followed by all the displacements which have just been reformed. The index for the whole of this set of displacements forms the contents of the Maxdof list.

10.1.5.1 Before the displacements of internal freedoms can be evaluated the external displacements must be re-ordered to the positions they occupied before being moved to accommodate summation with the structure which had been that far assembled. These displacements must also be resolved into the local axes to be compatible with the internal displacements which never had any need to be in any alignment other than the assembly orientation.

10.2 Obtaining the panel's displacement field.

10.2.1 The process of ordering the displacement list is by comparing the KP list for the current panel, which was formed during MOVE, with the Maxdof list. The essential thing is that the Maxdof list is in a random order when compared with the order of scanning the panel.

10.2.1.2 A simple pair of nested loops is set up to scan the whole of the Maxdof list for each location of the KP list examined. All the values of the KP list are contained within the Maxdof list plus some other values. When the current value of the KP list is found in the Maxdof list, the corresponding displacement value in the M3 list is found and copied into the next vacant location of a new list, the MOT list which is formed immediately after the M3 list. Upon completion of these nested loops the displacements in the MOT list have as an index the KP list. The M3 list is maintained intact to provide values to start the reconstruction of the next panel.



Location of lists referred to in 10.2.1

Fig. 10.2

10.2.2 The MOT list of external displacements is now in the order in which the panel was scanned after TRANS modified the reference axes of selected parts of the local stiffness matrix as directed by the KN list produced by PARTIT.

10.2.2.1 Those rows and columns of internal degrees of freedom of the overall local stiffness matrix which were removed, were removed by modifying other rows and columns all of which were originally referred to local axes. To calculate the displacements of those missing internal degrees of freedom requires the now known external and pseudo external displacements to be also given in relation to the panel's local axes.

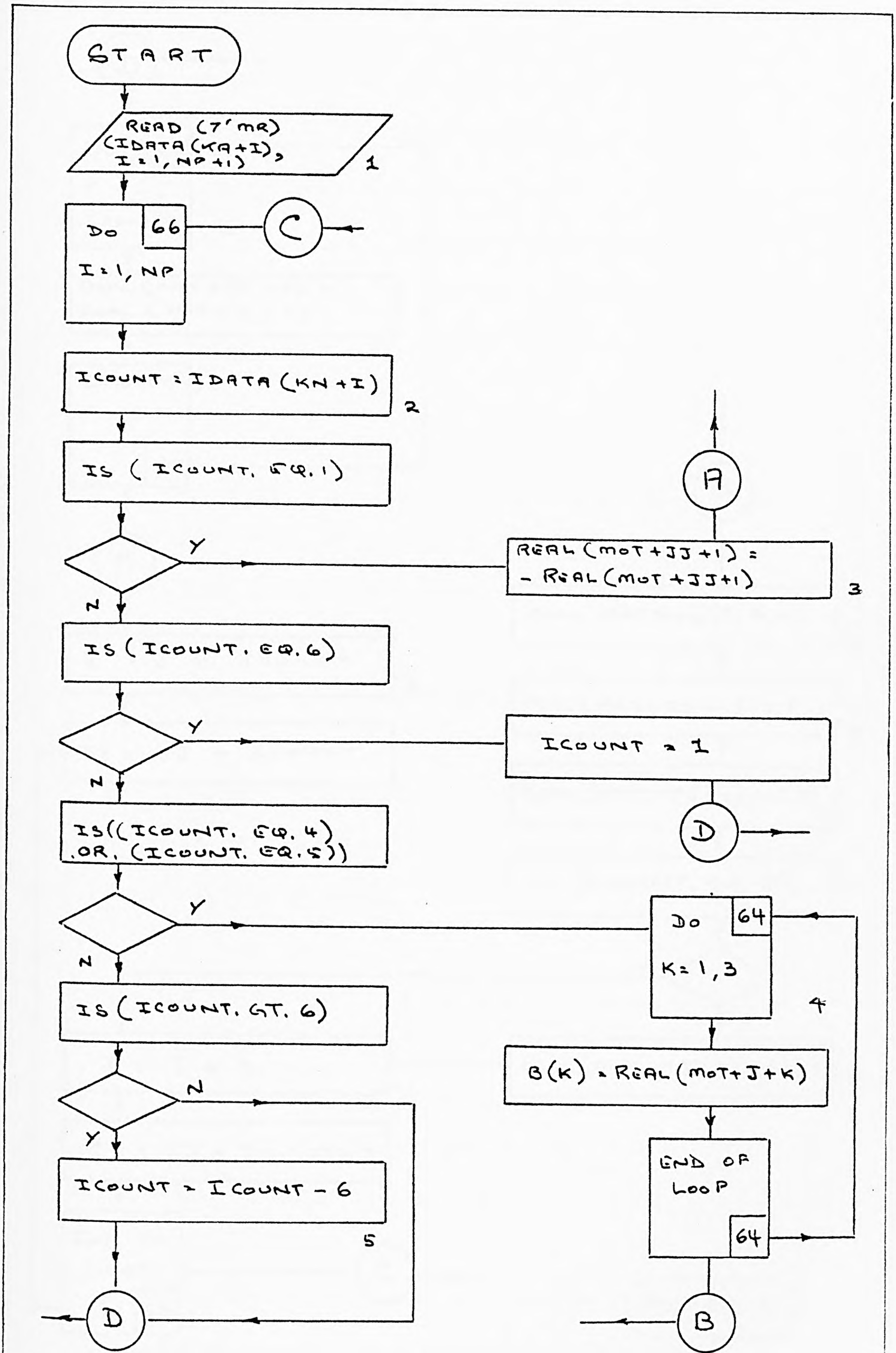
10.2.2.2 To convert the MOT list of displacements, some of which are already related to the local axes to all local axes orientation, reference must be make to the KN list and a process of reversing the action of TRANS applied. Only when a group numbered 4 or 5 is found in the KN list is action other than moving values to the left needed. If the group number is a 4 only the two local in plane displacements are needed from the three global values. A group numbered 5 will obtain the three orthogonal local displacements. There is one other tiny group, this is where the rotational freedom at the end of a stiffener was not suitably aligned to have its stiffness matrix row added directly to the adjacent rotational freedom on the next panel. In this case the single value group is numbered 1 and this will cause the sign of the rotational freedom to be changed. The KN list is recovered from disc at the same time as the recovery of the KP list.

10.2.2.3 In no case can the list derived from applying the reverse action of TRANS to the MOT list be larger than the MOT list. So that if the routine is applied systematically from one end to the other, the modified results may be stored by overwriting the MOT list. A flow chart showing the forming of a new MOT list of displacements relative to the panel's local axes is presented in Fig. 10.3.

10.2.3 At the conclusion of this stage all the displacements are in the order in which their equations were at the end of ASSEMB. The displacements of the internal freedoms may now be generated.

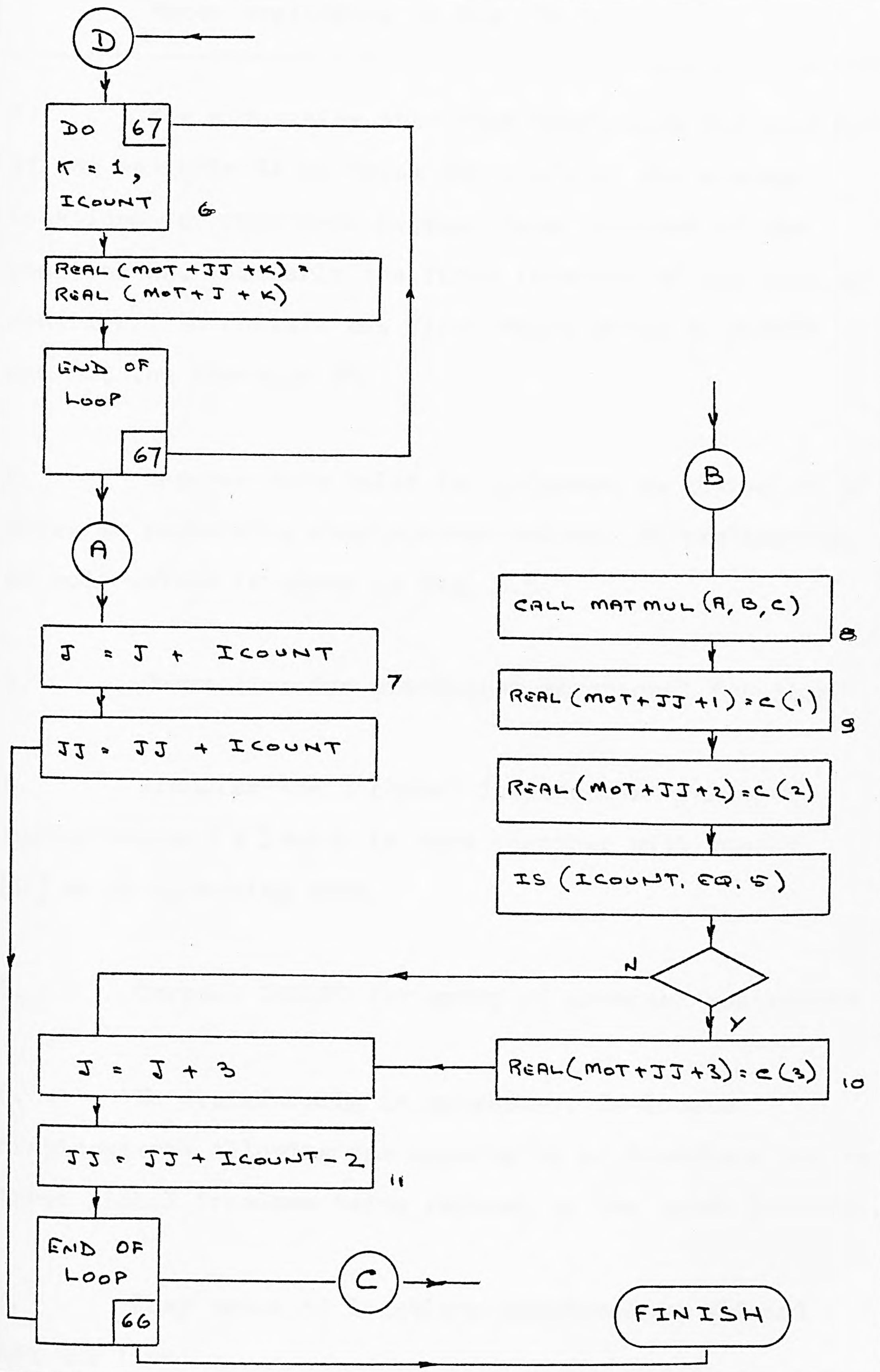
10.2.3.1 The routine to find internal displacements works backwards through the panel, since the last internal freedom to be removed could have only been expressed in terms of all those now considered active and which are known and have just been correctly ordered. Substituting all these known displacements into that last equation produced by REDUCE will give the value of the last internal displacement. The internal displacement before this was expressed in terms of all the actives, i.e. the external displacements and that internal value which has just been found. It can be seen that as each internal displacement is found it is in turn used with the others to find the next internal displacement.

10.2.3.2 As seen in the last paragraph the calculation of internal displacements requires a knowledge of the coefficients of the equations produced by REDUCE and subsequently stored on disc either at the end of the



Flow chart showing the reverse operation of Trans to obtain local displacements. Part 1

Fig. 10.3(a)



Flow chart showing the reverse operation of Trans to obtain local displacements. Part 2

Fig. 10.3 (b)

Notes applicable to Fig. 10.3

1. For simplicity this flow chart uses the position of the variable KA as being the start of the storage locations for recovered integer data, instead of the position KN. Similarly the first location of the data is considered to contain the first value given to ICOUNT and not the variable NP.

2. Recover code value for guidance on action to be taken on converting displacement valves. An explanation of code values is given in Fig. 8.5.

3. Correction for misaligned rotational freedoms.

4. Transfer the 3 global displacement values to buffer vector [B] which is used together with vector [C] as an operating area.

5. Correct ICOUNT for group of internal restraints.

6. No transforming is necessary. Re-locate displacements allowing for closing up of locations due to three global freedoms being reduced to two local freedoms.

7. Keep count of locations processed in old and new MOT list.

8. Calculate local displacements from global values, where [A] is the direction cosine matrix.

9. Return results to MOT list from buffer vector [C].

10. In the case of a node having three local freedoms recover local z freedom.

11. Correct new list pointer. Increase by 2 if ICOUNT is 4 and by 3 if ICOUNT is 5.

processing of that panel or when the available core space was fully filled. It is therefore necessary to recover these coefficients.

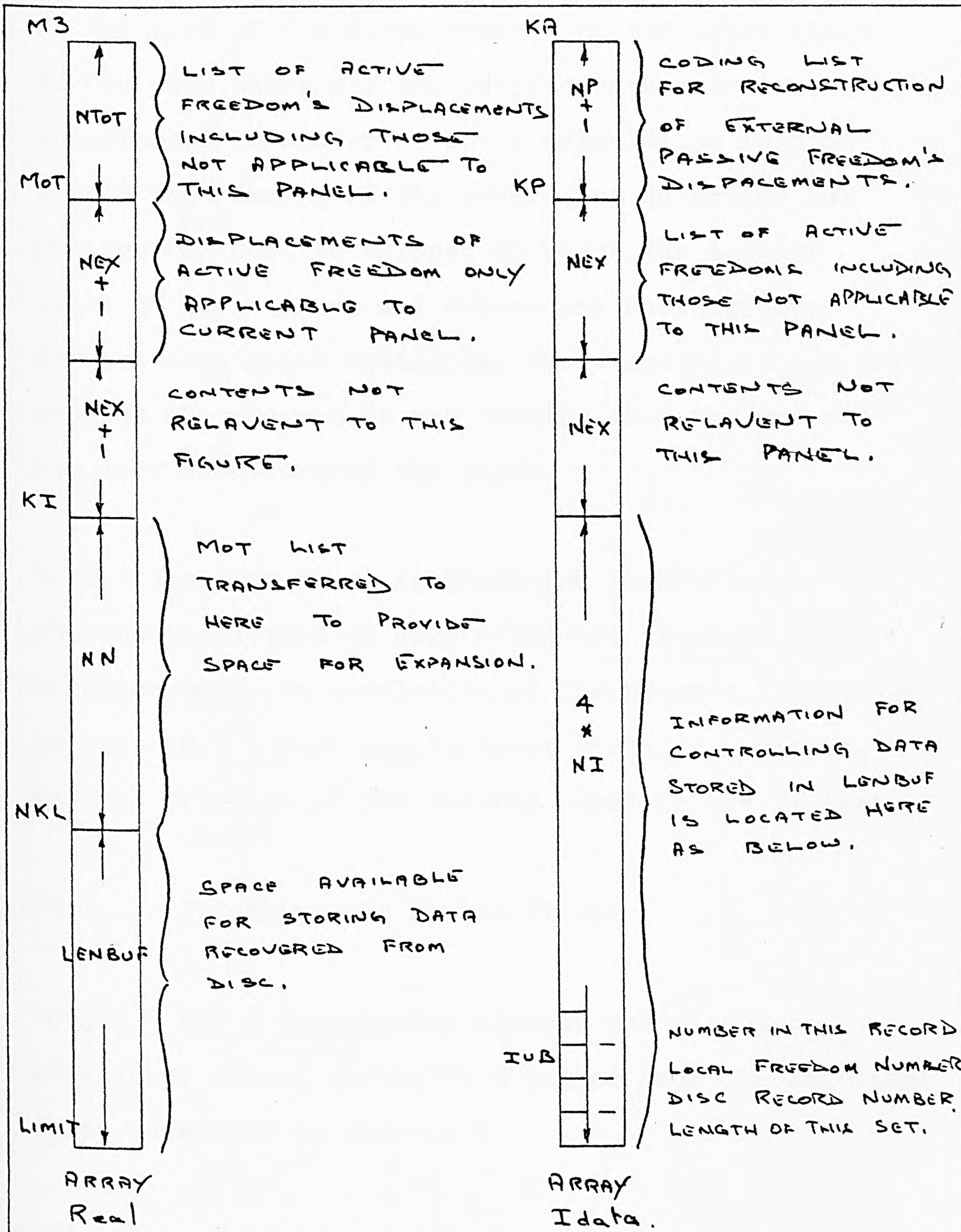


Chart showing the locations and variable positions used in Fig. 10.5

Fig. 10.4

To reduce as far as possible the time loss at the core / disc interface, equation coefficients are brought back from disc in as large a transfer as possible and stored straight behind the location of the last of the displacements of the total field. In the case where all the coefficients cannot be transferred in one operation a calculation is made within the running of the program to determine how many records must be skipped to reach the largest block of the current and subsequent coefficients for the core space available. The skipping is necessary because the program is now working from the end of the disc back towards the start.

10.2.3.3 A flow chart is presented at Fig. 10.5 showing the method of completing the internal displacements. On completion of the program illustrated in Fig. 10.5 a full displacement field is available for all freedoms of the current panel in the KI list.

10.3 Obtaining the stress field.

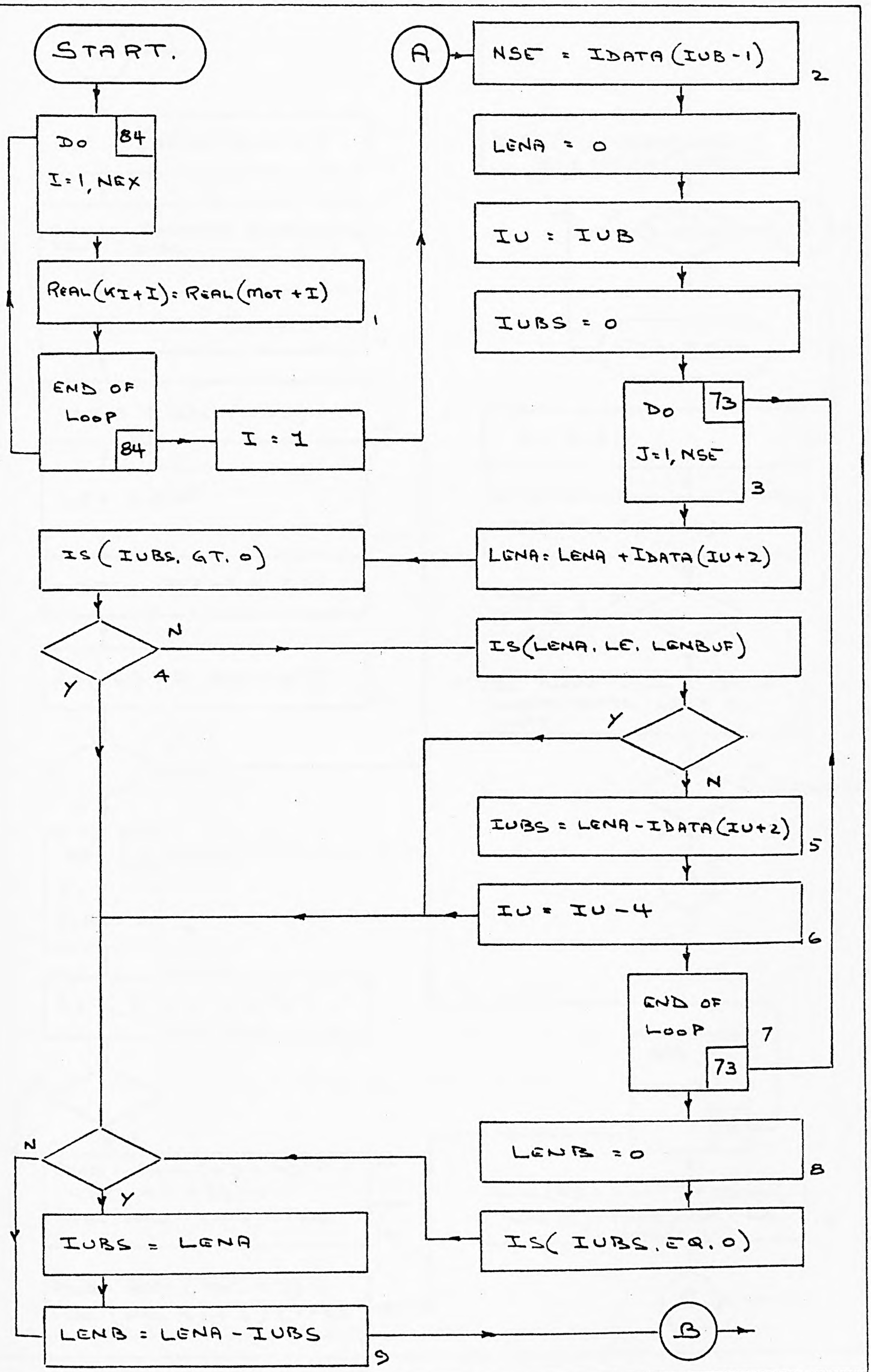
10.3.1 For a rectangular element the elastic structural stress vector is obtained from the relationships developed in chapter 4.

$$\sigma = [D] \; e \qquad \ldots(4.8)$$

where $$e = [B] \; a \qquad \ldots(4.35)$$

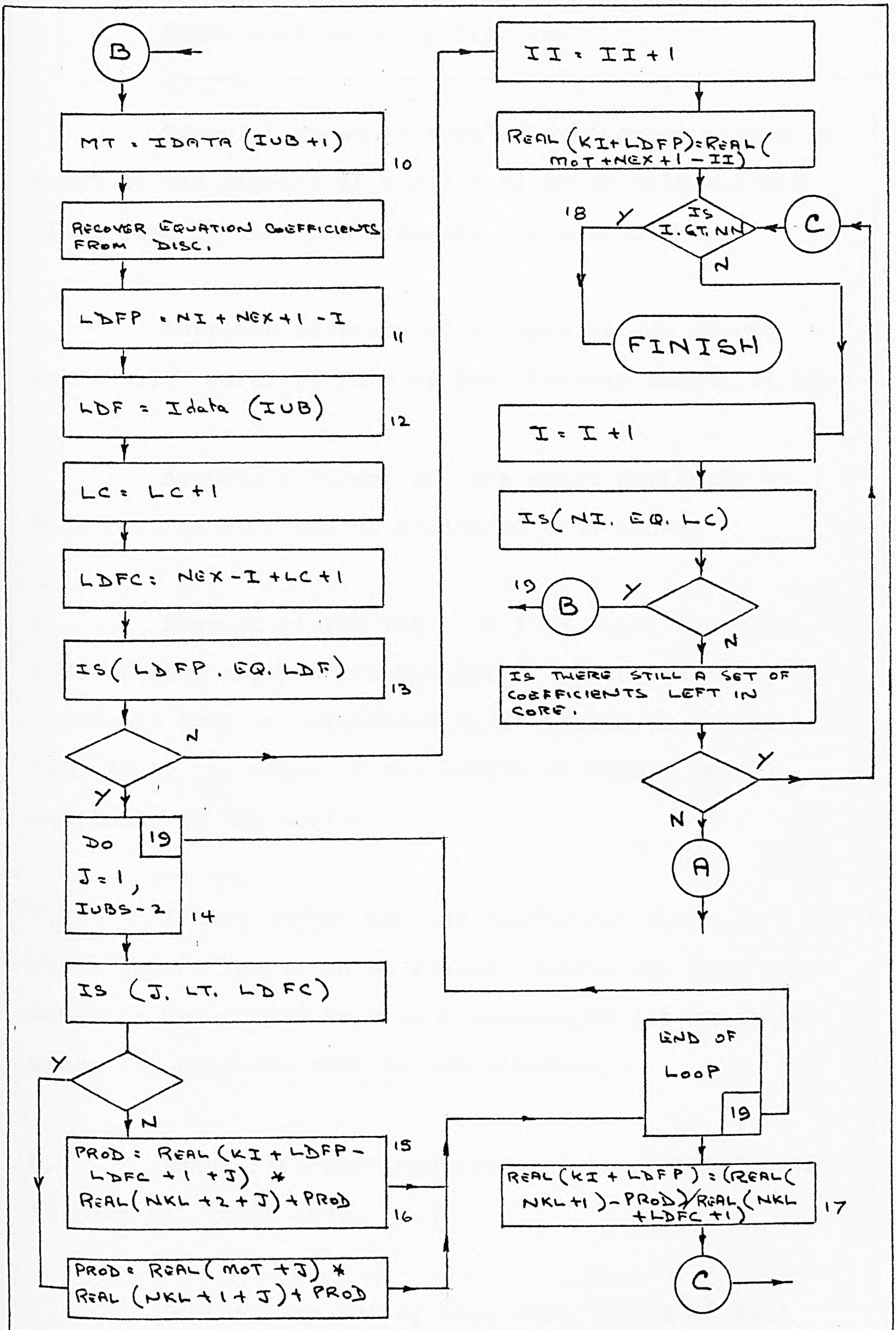
and $$u = [A] \; a \qquad \ldots(4.33)$$

hence $$\sigma = [D] \, [B] \, [A]^{-1} \; u \qquad \ldots(9.2)$$



Flow chart showing method of completing internal displacements. Part 1

Fig. 10.5(a)



Flow chart showing the method of completing internal displacements. Part 2

Fig. 10.5(b)

Notes applicable to Fig. 10.5
1. Transfer known external displacement values to start of the segment of array Real which will contain final list of all displacements for this panel.
2. Initiate recovery of integer values stored in Idata which refer to sets of coefficients stored on disc.
3. Ascertain amount of core space available in Real for the storeage of recovered equations.
4. Prevent stored value of IUBS being increased. IUBS remains empty until the buffer space available for recovered data is calculated to be filled by J equations. IUBS takes the value of the length of usable space available in the buffer.
5. If the buffer has not sufficient space left for whole record length to be stored, delete the last value added to bring IUBS back to a measure of the available space for complete sets of coefficients.
6. Reset reference numbers of sets of coefficients to next set to be seen.
7. Do not jump out of loop when length of data that can be recovered from disc is known. The full number of loops must be completed to obtain details of amount of disc space to be skipped when reading from disc

in a forward direction, noting that counting of records has been recovered in a reverse direction.

8. If IUBS is still set at zero then all the coefficients on the current disc record can be recovered and stored in Real.

9. If all the coefficients cannot be recovered at one call, derive amount of disc that has to be skipped. If all can be recovered LENB will be set to zero.

10. Using data recovered from Idata find relevant disc record number.

11. Set up integer for local degree of freedom position relative to variable KI.

12. Recover from Idata the number of the next local degree of freedom whose coefficients are to be examined.

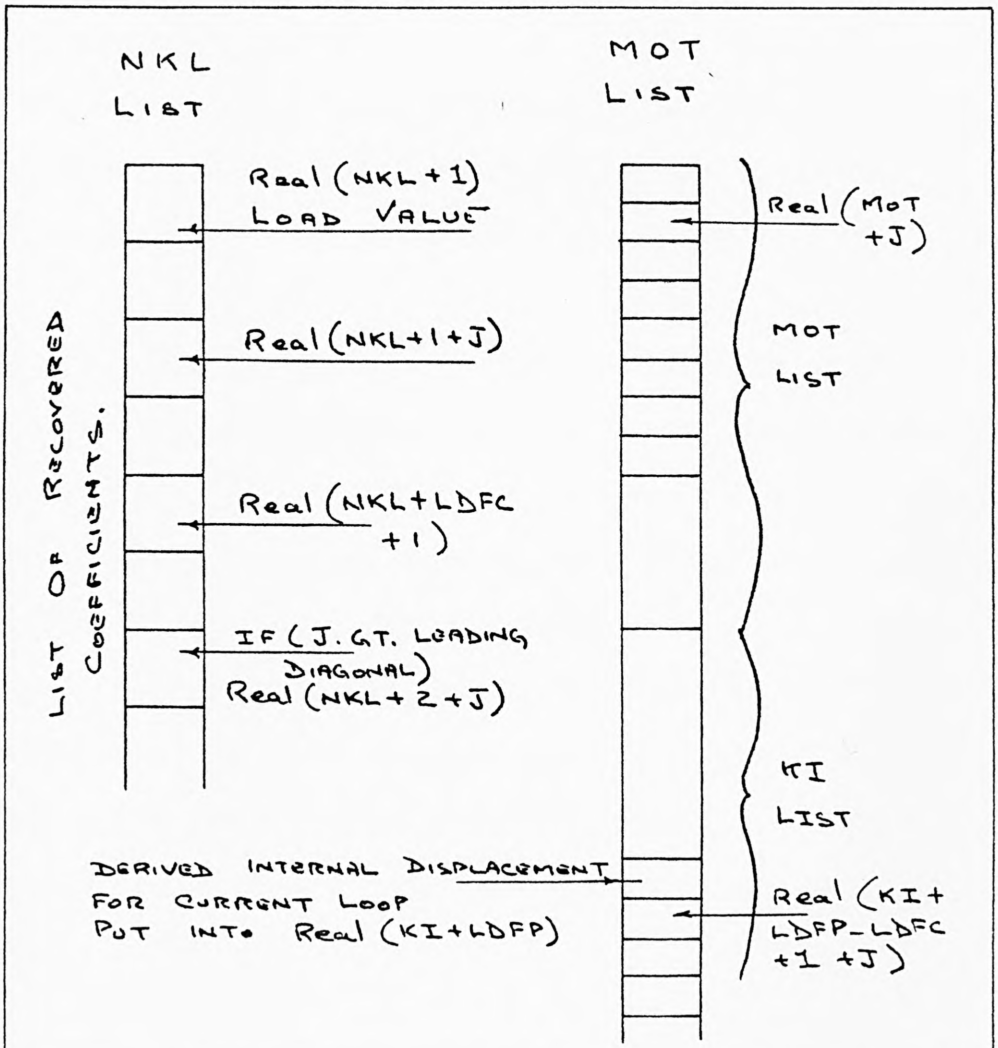
13. Compare the recovered local freedom number of the next freedom to be inserted with the local count through the list being assembled.

14. Start process to calculate the value of next internal displacement and insert it at the lowest vacant space in the KI list.

15. Variable LDFC maintains a count through the coefficients to have available the knowledge of whether

or not this location is before the leading diagonal term.

16. Summing the products of all available coefficients and their applicable displacements. If the coefficient and displacement refer to freedom processed by REDUCE before the leading diagonal term the MOT and NKL lists are used, otherwise the KI list and the calculated or relocated displacement values are used. Fig. 10.5 shows the storage of these lists.



RELATIVE STORAGE LOCATIONS AND VARIABLE NAMES FOR CALCULATING AND STORING THE DISPLACEMENT OF AN INTERNAL FREEDOM.

FIG. 10.5 (c)

17. Calculate and add next internal displacement to KI list.

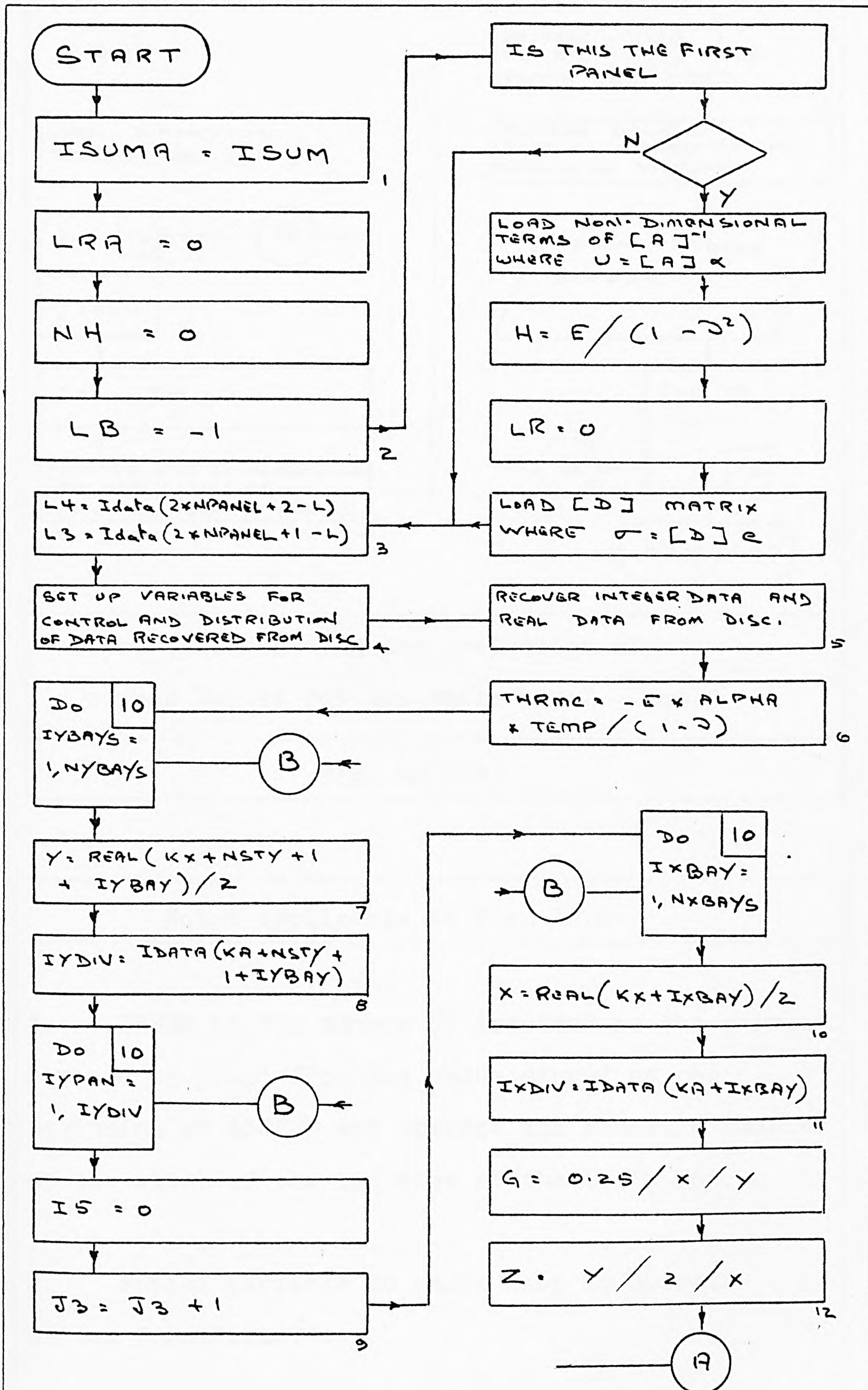
18. NN is variable for total number of values to appear in KI list.

19. If all internal displacements have been formed operate only on known displacements.

10.3.1.1 The matrix multiplication shown for the stress vector evaluation is made at each node for which that node is at the bottom left hand corner of the current element. Consequently the last node of each row is missed as is the whole of the top row. Though the matrix multiplication above must be made once for every rectangular element, (see the Flow Chart, Fig. 10.9), this is not necessary for the formation of the individual matrices. The array [D] is formed once for the first panel as are the non-dimensional terms in the $[A]^{-1}$ array. The dimensional terms of $[A]^{-1}$ can be left unchanged between vertical stiffeners since the element sizes are constant between stiffeners. (See the Flow Chart at Fig. 10.6).

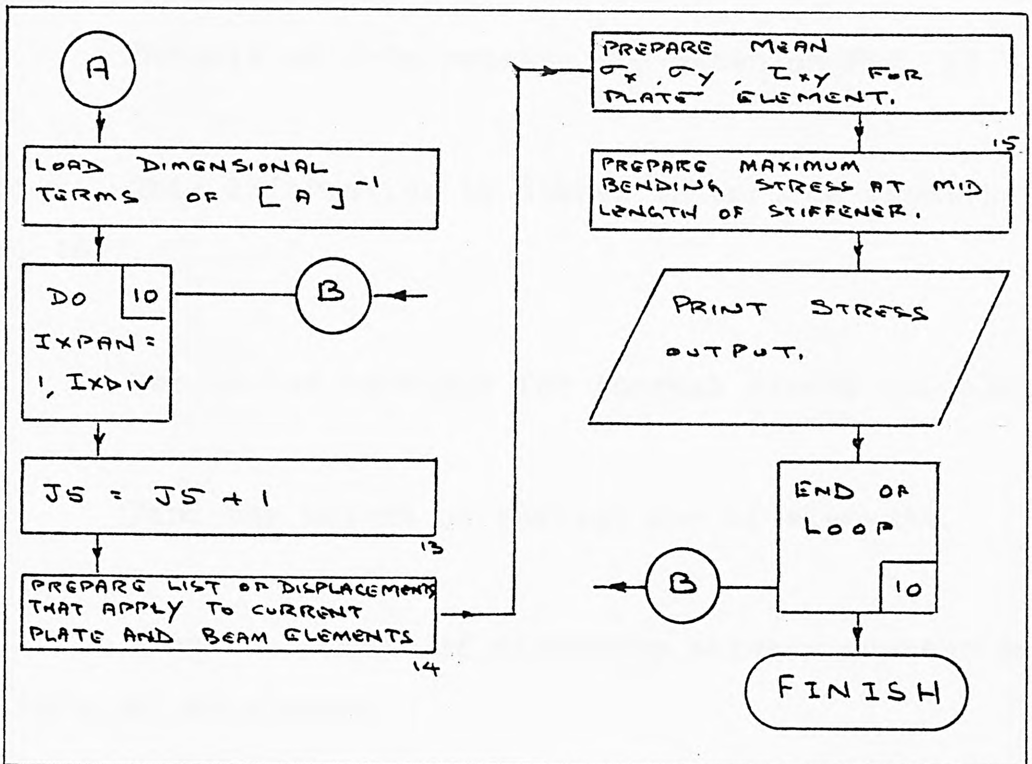
10.3.1.2 The displacement vector obviously will change at each element. Section 10.4 describes the formation of the displacement vector and the recovery of beam displacements. Since the stress is not constant in the type of rectangular elements used, a stress vector is generated at each corner and a mean stress is printed. Because of this the [B] matrix will have a different value at each corner. A description of the evaluation of stresses is contained in section 10.5.

10.3.2 At the start of processing of each panel after the final inversion it is necessary to make a call to disc to recover details of the number of elements and their sizes, the types of stiffeners



Flow chart showing the production of stress values for the whole panel. Part 1

Fig. 10.6(a)



Flow chart showing the production of stress values for the whole panel. Part 2

Fig. 10.6(b)

Notes applicable to Fig. 10.6

1. ISUMA is the number of freedoms in the current row and is found from the value stored at the beginning of ASSEMB and presets the result's counter at the start of the top edge of the first row.

2. Preset variable to cause loop to operate at the first time.

3. Set up variables to control data recovery from disc.

4. Details of this process are shown on Fig. 10.7.

5. This information is stored in core as shown in Fig. 10.7.

6. Set up the constant for thermal stress calculations.

7. Find the height of current row of elements.

8. Find the number of divisions between current pair of vertical stiffeners.

9. Update count of number of divisions along local y side of current row.

10. Find width of elements in block between current pair of vertical stiffeners.

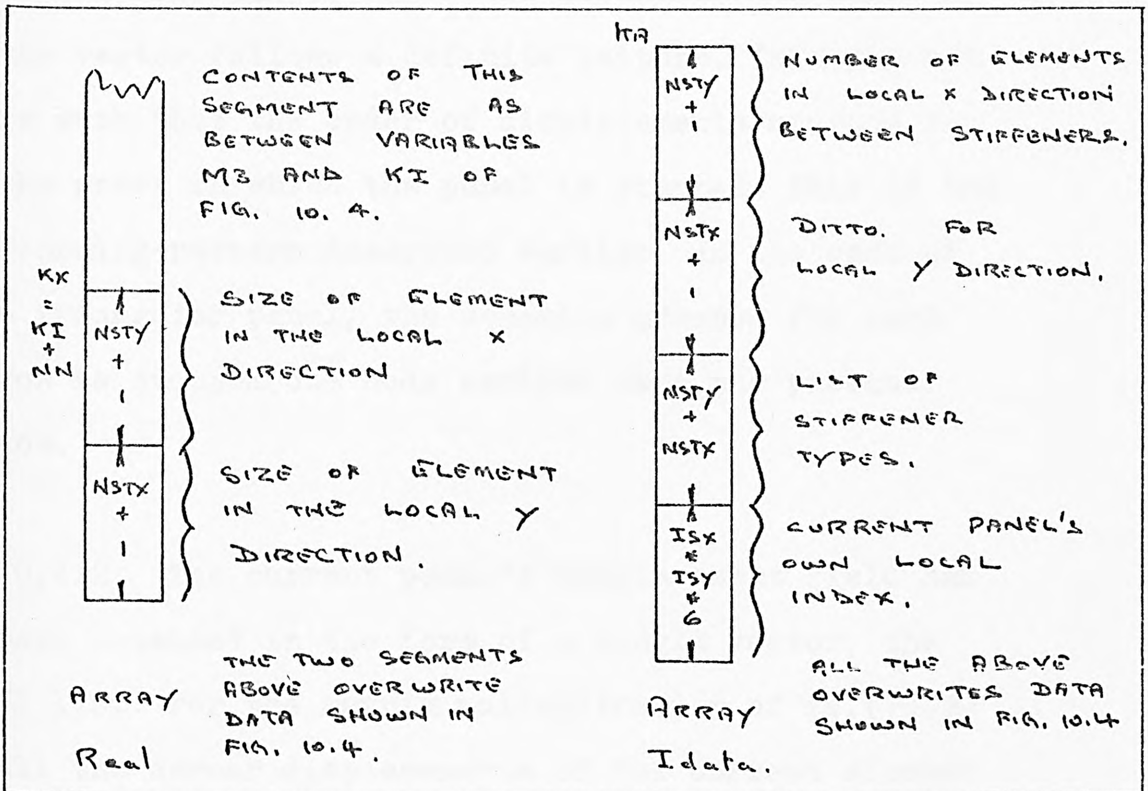
11. Find number of elements in current block.

12. Introduce new variable to prevent repetitive calculations.

13 Update count of number of divisions along local x side of current column.

14. Details of this process are shown in Fig. 10.9.

on the current panel and the list of the degrees of freedom numbering, Fig. 10.7 shows their storage locations.



Storage locations of data recovered from disc at start of stress calculations.

Fig. 10.7

There is no need to know the actual number allocated to a freedom, it is sufficient to know that there is a freedom existing at that location. This knowledge could also be obtained not by any reference to lists of freedom numbers but by the state of the triggers indicating the presence of horizontal and vertical stiffeners. These triggers are automatically set to zero if the loop counters are not at the first bay loop after starting a new bay between stiffeners.

10.4 Obtaining displacement values of current

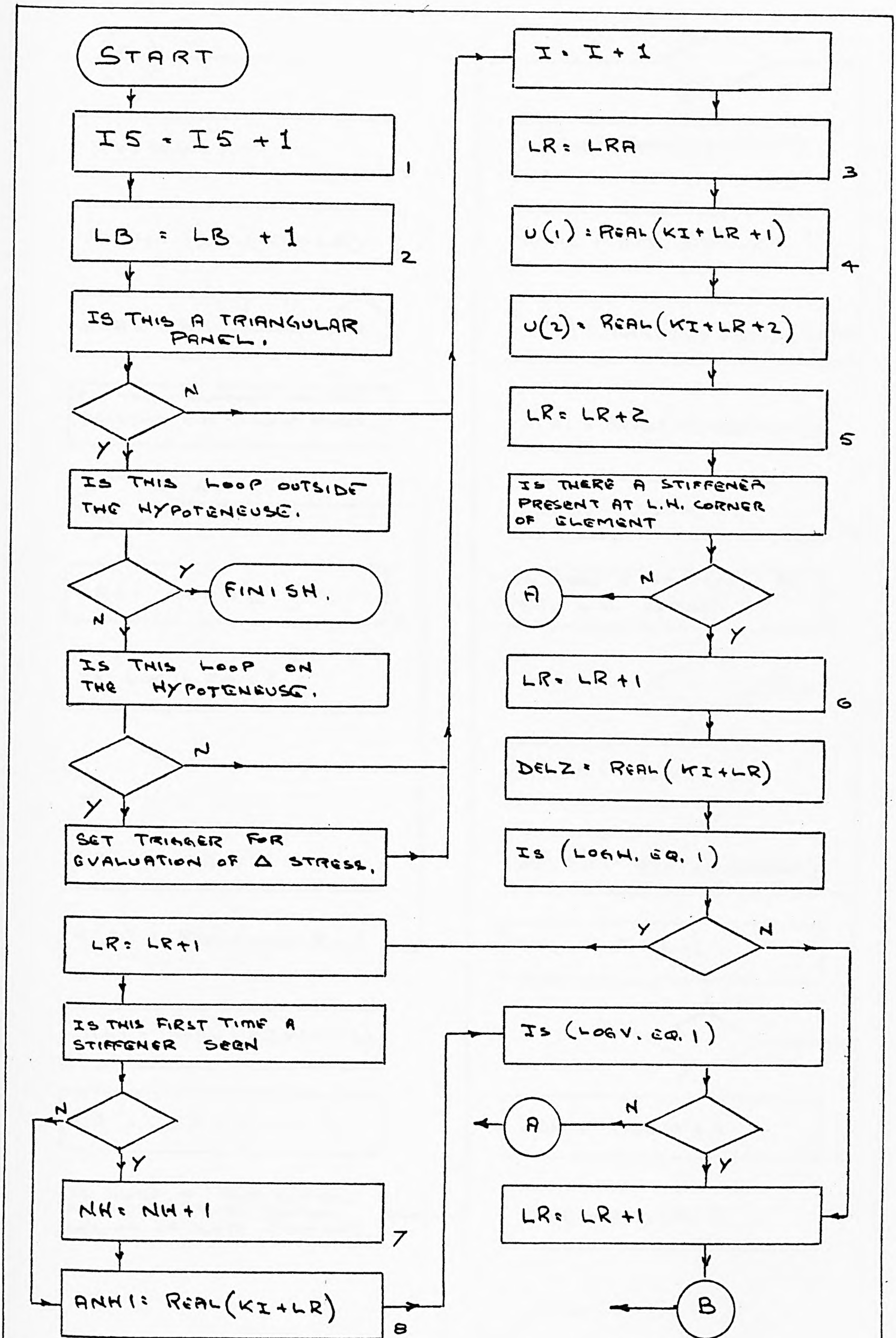
elements.

10.4.1 There is now no need to maintain an index of the freedoms to which the displacements apply as the vector follows a definite pattern. This pattern is such that the order of displacements storage is the order in which the panel is scanned. This is the scanning pattern described earlier. In the case of a triangular panel, the scanning process for each row is stopped one node earlier than the previous row.

10.4.2 The current panel's displacement field has been obtained in the form of a single vector, the KI list. For the matrix multiplication of Eq.(10.2) all the corner displacements of the current element must be assembled into a small vector $[U]$ and in the order shown in Fig. 4.5 for a rectangular element or Fig. 4.17 for a triangular element. For a beam element displacements along the longitudinal axis are the same as certain of the associated plate's freedoms. The beam displacements which do not appear in the vector of results for a plate element are given variable names. The flow chart of Fig. 10.8 shows how the displacement values are selected.
It is necessary to maintain two counters for the progress of selection of values from the KI list. Different counts are required along the top and bottom edges of a row of plate elements. This is because of the combination of the presence of a horizontal

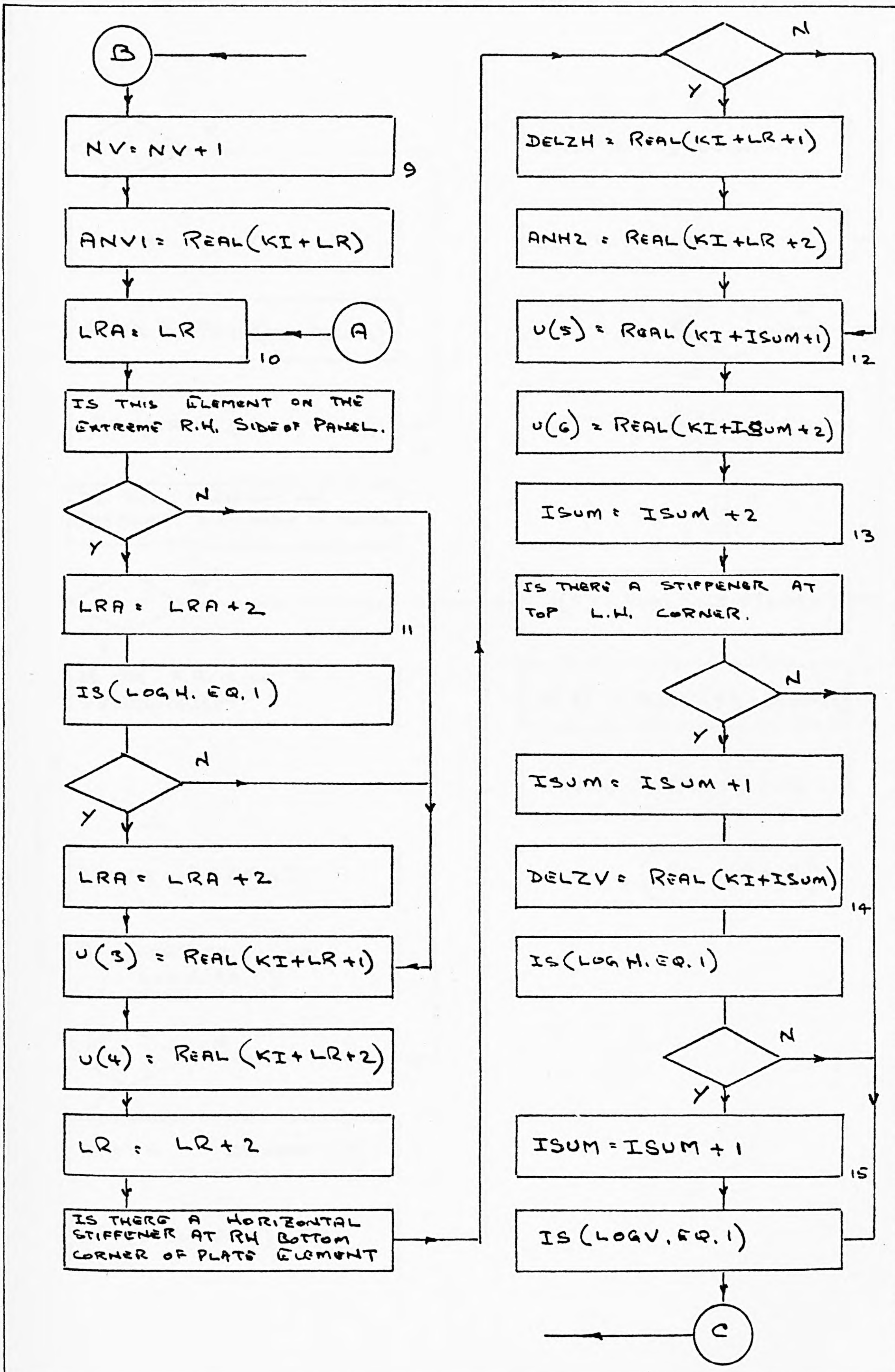
stiffener along one side of a row of plate elements with the absence of a stiffener along the parallel side will mean that there will be two extra displacement per element along one side resulting from the freedoms of the stiffener, causing a different rate of movement through the results.

10.4.3 The lower edge count starts initially at zero since the displacement sought at this stage is at the begining of the results. The upper edge count must be primed and this is done by a variable, the value of which is found during this panel's pre-compression stage, being put off with other data to disc and now being recovered during the disc call by DISP. As these two couters move across the panel they are increased for each displacement value seen. When the left hand corner results have been transfered to the [U] vector the value of the counters then existing are stored, because these which now indicate the position of the right hand corner results are also indicating, since they are the same, the position of the results for the next element's left hand corner. The counter's value for the left hand corners of the second panel would be lost if there was a stiffener present causing the counter to be increased until it indicated the right hand corner of the second element, therefore the counter value must be stored before any possible stiffener displacements are read. At the right hand side of a panel the low counter's value becomes the



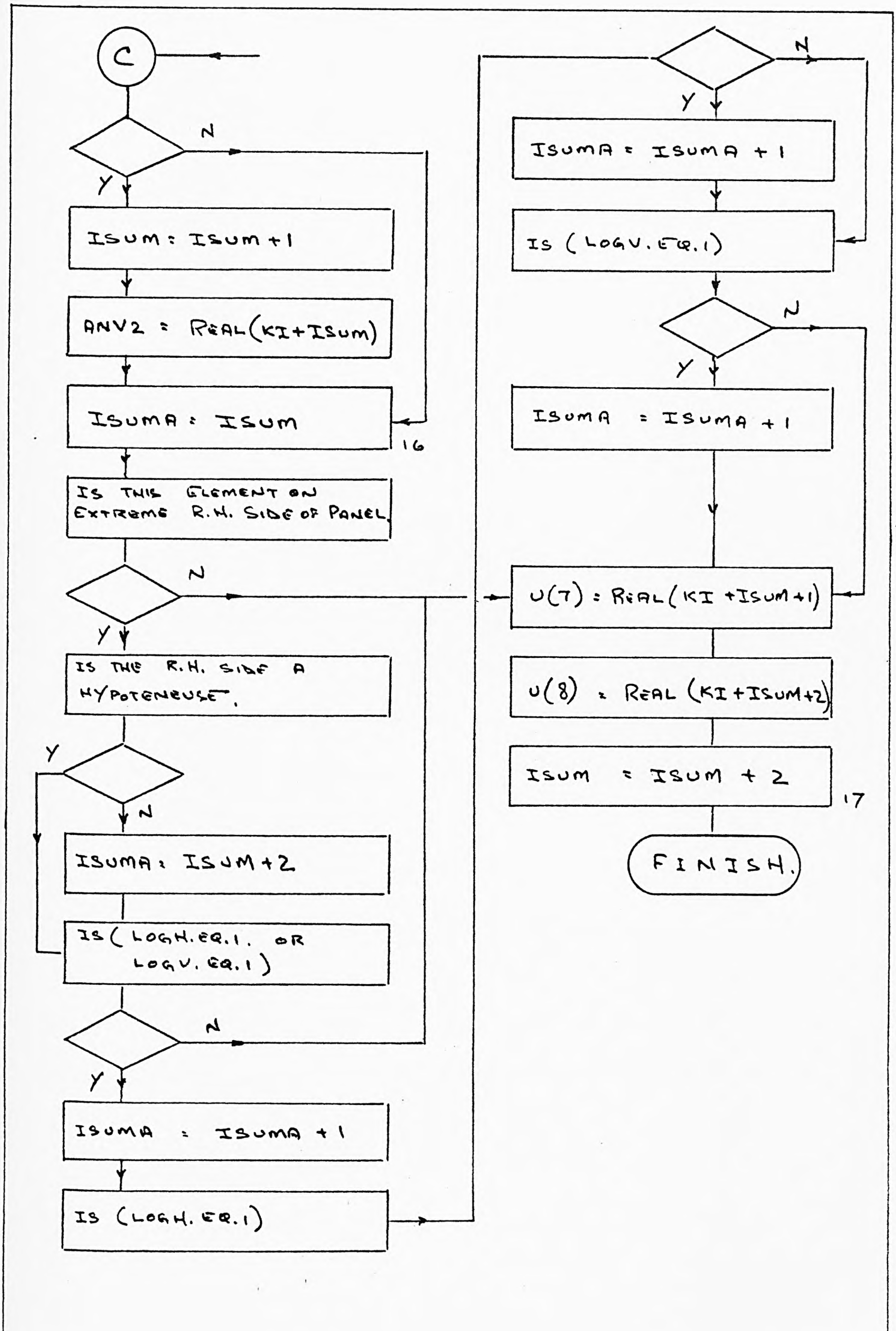
Flow chart for obtaining the displacement values for the current element. Part 1

Fig. 10.8(a)



Flow chart for obtaining the displacement values for the current element. Part 2

Fig. 10.8(b)



Flow chart for obtaining the displacement values for the current element. Part 3

Fig. 10.8(c)

Notes applicable to Fig. 10.8

1. Triangular panels are scanned as rectangular panels. A variable I5 is kept updated to indicate the start of the domain outside of a triangular panel but still within the area of a rectangular scan.

2. Update row counter.

3. At the start of each element recover count through list of displacement results from right hand lower edge of last element to left hand lower edge of current element.

4. Load part of displacement vector as it applies to bottom left hand corner of current element.

5. LR maintains a count through list of displacements along the lower edge of a row of elements.

6. Update LR count in the case of a third order freedom due to a stiffener being present at the bottom left hand corner node of current plate element.

7. Update count of horizontal stiffener types.

8. Store rotational displacement value of left hand end of horizontal stiffener.

9. Update count of vertical stiffener types.

10. Keep present value of lower edge count now at right hand side of element for use on left hand side of next element. Transfer current count value to new variable for counting through results applying to right hand lower corner.

11. Update count of plate freedom displacements when there is a horizontal stiffener present.

12. Load displacement vector for top left hand corner node of plate element.

13. Increase count through displacements for freedoms along top edge of row of elements.

14. Store third order freedom's displacement for node at the top left hand corner, when there is a stiffener present.

15. This takes account of a dispalcement value in the list of results but the result is ignored until the horizontal stiffener to which it applies lies along the lower edge of the row of plate elements.

16. Apply to count of results along top edge of plate elements similar storing arrangement as applied to lower edge as detailed in note 10.

17. This is final update of results count before starting at first freedom of next node.

lower counter for the next higher row of elements after it has been corrected for the displacement values of the right hand corner of the last element which explained above would normally be skipped. Similarly the upper counter at the far right hand side is modified to become the upper counter of the next row at the far left hand side.

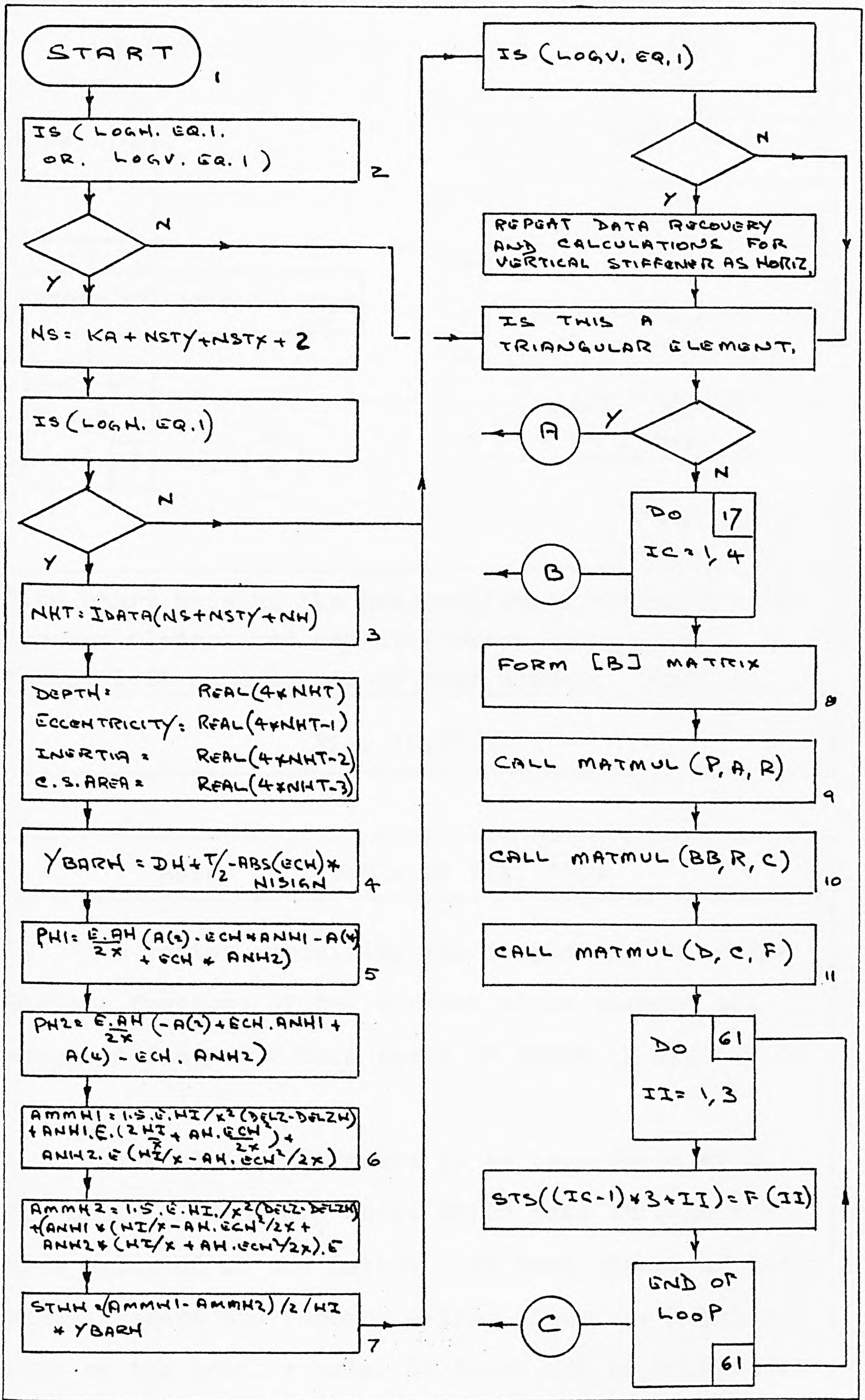
10.4.4 Two other counters are kept running at this time to take account of the stiffener type being seen. As the panel is scanned across, the vertical stiffener type is continually updated at the start of each new bay, but is brought back to zero at the start of a new row. The horizontal stiffener count is held constant for a transverse scan since the stiffener properties must remain constant along their length and is only updated at the start of each new horizontal bay.

10.5 Obtaining the stress output.

10.5.1 Rectangular panel stress derivation.

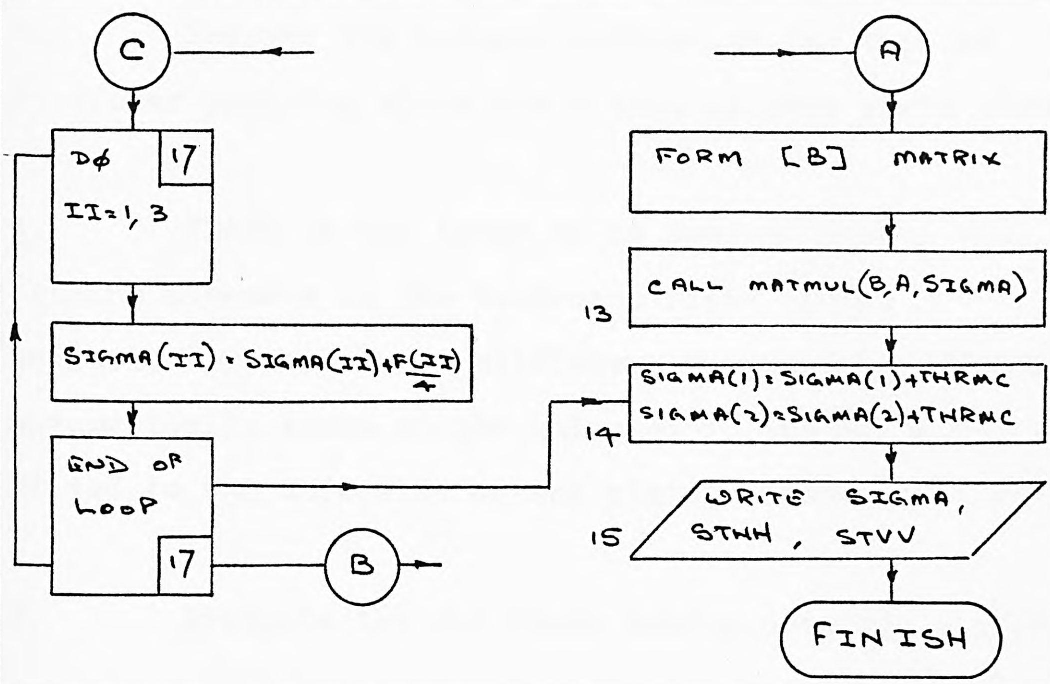
10.5.1.1 Using Eq. (4.8) the set of stresses existing at a node may be found if the strain vector e is known.

10.5.1.2 Referring to Eq. (4.7) the relationship between the set of strains for a corner of a plate element and the displacements of all the corners



Flow chart showing the preparation of stress for the current element and any stiffeners being at the bottom left hand corner of that element. (Part 1).

Fig. 10.9(a)



Flow chart showing the preparation of stress for the current element and any stiffeners being at the bottom left hand corner of that element. (Part 2).

Fig. 10.9(b)

Notes applicable to Fig. 10.9

1. A vector containing the displacements of the inplane freedoms of the current plate element are prepared ready for this stage as shown in Fig. 10.8.

2. The plate stiffeners to be considered with any plate element are those which pass through the node situated at the bottom left hand corner of the plate element and running either along the local x axis or the local y axis. If there are no stiffeners present the routine jumps to consider the plate stresses.

3. Recover the integer indicating the type of stiffener applying along the x axis of this plate element.

4. YEARH is the lever to be used to obtain the bending stresses in the outermost fibre of the horizontally orientated stiffener. The variable NISIGN automatically takes on the value -1 if the stiffener is fitted to the underside of the plate, otherwise it is 1.

5. Evaluate the end loads coming onto the stiffener.

6. Evaluate the bending moment being applied to the stiffener.

7. Evaluate the maximum bending stress in the stiffener.

8. The [B] matrix is defined in Eq. (4.34).

9. Pre-multiply the displacement vector by the inverse of the [A] matrix as defined in Eq. (4.33), now stored in the vector P to obtain the a coefficients where e = [B] a. These a coefficients are stored in the vector R.

10. Obtain the strains occurring at the ICth corner of the element.

11. Obtain the stresses occurring at the ICth corner of the element. These are stored in the three location

vector F and are subsequently transferred to the vector STS which holds all the stresses at each corner of the current element. The contents of STS are printed out if there is a requirement to know the stresses on the element sides.

12. Mean the stresses occurring at each corner to give stress at plate centre.

13. In the case of a triangular plate element form the appropriate [B] matrix as defined in Eq. (4.59) and post multiply it by the displacement vector to give the plate's stress values.

14. Add the thermal stress values where THRMC is defined in Eq. (6.2).

15. The written stress output consists of the plate's mean direct and shear stresses from the vector SIGMA, the bending stress in the horizontal stiffener if any STHH, the bending stress in the vertical stiffener if any STVV. The maximum and minimum principal stresses and the angle of one of the principal planes.

of that element is by the matrix [b] , where [b] represents the exact strains due to unit displacements.

Consider the case where the rectangular element is derived from equations based upon a linear displacement of its side. Having decided upon the displacement vector as in 9.4, the set of strains is easily found by post multiplying the [b] matrix by the appropriate displacement vector. A mathematical definition of this [b] matrix is given in Eq. (4.21). There are eight void locations in this array, the sixteen with definite values are loaded by a standardized procedure via a small stage necessary to prime the procedure with the only two variables which change, being those representing the co-ordinates of the corner.

10.5.1.3 In the case where the rectangular element has been derived by recourse to distortion modes, unlike those for the last element the set of strains for a corner of this type of element is not related to the whole element's displacement vector by a single available matrix, but is connected as shown is Eq. (4.40) by the matrix product $[B] * [A]^{-1}$. The matrix [B] , which is defined in Eq. (4.35) contains only seven non zero terms out of an array of twenty four locations and is simply loaded once the appropriate values of the corner co-ordinates are known from using the same priming stage as used for the [b] matrix in 10.5.1.2. The $[A]^{-1}$ matrix contains twenty six values which have the absolute values of either a half or a quarter. This speeds the loading as these locations need to be loaded just once per structure. There are a further twenty six locations which remain void leaving only twelve locations dependent upon the size of the element and as these are not dependent upon co-ordinates, they need be changed only after a new group of elements between stiffeners is started and not for each element. For both types of element the set of strains for the current corner is stored in [e] overwriting the previous corners' strains.

10.5.1.4 The set of stresses at each corner of the element is found by post multiplying [D] by the set of strains just found for that corner as described above in either 10.5.1.2 or 10.5.1.3 from the previous corner. The stresses for a panel could be supplied directly as calculated by using Eq. (4.8) at each corner of each element. It was felt that this would give too many results. A more reasonable amount of information is presented if a mean value is chosen. This could be done by considering all the elements joining at a particualr corner and taking the average of their individual contributions. The way selected was to consider each element individually and using the stress results at the four corners obtain a linear mean at the centre of the rectangle. This was easily achieved by setting up a three location array SIGMA which had each location loaded with a quarter of the relevant stress obtained as each corner was in turn processed. The full set of stresses and strain found, for each corner is available if required by printing out the twelve location arrays STS and STR into which the three location arrays [F] and [C] are copied in sequence as they are formed for each corner in turn.

10.5.2 Triangular panel stress derivation.

10.5.2.1 In 4.3.1 it was explained that triangular panels are subdivided into and processed to maximise the number of rectangular elements into which a panel is idealized and that a triangular element is used only as the last element of a row in a triangular panel.

10.5.2.2 In generating the stress output for a triangular panel, the process proceeds as in 10.5.1 until the appearance as the current node of the node immediately before the hypoteneuse edge.

10.5.2.3 The assumed stress distribution in a triangular element is constant. Therefore only one calculation need be made for each such element. The averaging of a value as for rectangular elements is not required. As before substituting Eq. (4.7) into Eq. (4.8) gives

$$\sigma = [D] * [b] * u \qquad \ldots(10.3)$$

Post multiplying [D] as defined in Eq. (4.59) gives

$$\frac{E}{2A(1-\nu^2)}\begin{bmatrix} x_{32} & -\nu y_{32} & 0 & \nu y_{31} & x_{21} & 0 \\ \nu x_{32} & -y_{32} & 0 & y_{31} & \nu x_{21} & 0 \\ -y_{32}\left(\frac{1-\nu}{2}\right) & x_{32}\left(\frac{1-\nu}{2}\right) & y_{21}\left(\frac{1-\nu}{2}\right) & 0 & 0 & x_{21}\left(\frac{1-\nu}{2}\right) \end{bmatrix} \quad \ldots(10.4)$$

The matrix product $[D] * [b]$ as defined in Eq. (10.4) is stored directly in the program. This product is dependent upon the length of the sides of the element and therefore needs to be loaded only once per group of elements between stiffeners. The program uses the same locations to store Eq. (10.4) as is used to store $[b]$ matrix of the rectangular elements which would in any case be redefined at the start of a new row which must occur after a triangular element, except when the apex of a triangular panel is reached. The displacement vector for a triangular element should contain only six locations. In our case the same vector space is used and eight values loaded, but the last two are ignored by the subsequent matrix multiplication. The matrix product of Eq. (10.4) and the displacement vector is written directly into the SIGMA array ready for printing.

10.5.3 Stiffener bending stress derivation.

10.5.3.1 The matrix governing the relationship between a beam's end loadings and the resulting displacements is for this analysis given in Eq. (4.79).

10.5.3.2 To obtain the bending moment being applied to the ends of the stiffener may be found by multiplying out the appropriate lines of Eq. (4.79) with applicable displacements obtained from the KI list. In the case of plate elements a vector of displacements was obtained. In this case two values of the vector produced for the associated plate element are used as the first and fourth beam displacement values. Which two values are used depends upon the orientation of the beam. In the case of a horizontal beam the plate's second and fourth displacements are used. If the beam is vertical then the plate's first and fifth displacements apply.

The out of plane and rotational displacements are found at the same time as the plate's displacement vector is loaded. Details of this are given in 10.4. To complete the values of the beam's stiffness matrix applying at this stage details of the beam's properties are recovered from permanent core locations after being found using particulars brought back from disc at the begining of the current panel's displacement assessment. The stiffener's depth and eccentricity are compared to evaluate the maximum distance from its neutral axis to the outer fibre. Engineer's bending theory is used to give a maximum bending stress for the stiffener.

10.6 Thermal stresses.

From Eq. (4.6) we see that the component of total stress due to thermal loadings is given by $\alpha T [D_T]$ where $[D_T]$ is defined in Eq. (4.5). It will be seen from Eq. (4.5) that the values of the vector of thermal stresses are independent of the position on the panel at which the stresses are being measured. There is no facility within the handling of any one panel to cater for more than one temperature variation. This restriction is considered satisfactory. If parts of a panel receive different temperature changes, then this loading is accommodated by dividing the panel into smaller panels, so that each panel receives a consant temperature change. At the beginning of the stress analysis for each panel the constant term of Eq. (4.5) can be evaluated immediately the current panel's details have been recovered from disc. No further reference is made to thermal stresses until the mean direct stresses occuring at the element's centre are calculated at which time the thermal stresses are added to the SIGMA stress vector.

11. Conclusions and Future Work.

11.1 Initial objectives.

11.1.1 Referring to 2.2.1 it can be seen that the objectives of this thesis were:

1. The solution by finite element analysis of structural problems where the stiffness matrix exceeds the core available.

2. A reduction in the amount of data to be prepared by hand.

3. No loss of accuracy.

4. No increase and if possible a reduction in computer time.

11.1.2 The running of comparisons in time and core space between packages is difficult since even if they use the same type of elements there will be differences in the time taken to process the input data, depending on the degree of data preparation by hand before the run and the core space required for the particular package. These problems would be further aggravated if the runs were made on different computers. Where comparisons of time are needed all the runs have been made on the City University's Honeywill 66/60 computer. Only two of the test cases contained in appendix C were not run on this computer.

11.1.3 The following shows how all the above objectives have been very well achieved.

11.2 Final achievements in relation to 11.1.1.1.

11.2.1 In considering 11.1.1.1 example 5 of appendix C uses an idealization of a box beam with 923 nodes having 2104 real degrees of freedom as opposed to those extra degrees of freedom which appear automatically with some packages and then require to be suppressed as unwanted out of plane freedoms. The size of the real number array used for storing the representation of the structural stiffness matrix, the load vector and other data is only 12K. A further 2K is used to store the various indices and integer data. The complete package of program data generating routines including those not used for this problem, the integer array and the real array using double precision occupied just under 64K. This problem proved far too large for either Lusas or Sap IV to deal with, when they were constrained to the limitations of core and time applied to them by the City University's Computer Department of 95K for Sap IV and 120K for Lusas. As commented upon in 11.1.2 these comparisons are by themselves rather dubious. Appendix C contains details of the largest analyses which were successfully run using Lusas and Sap IV. It can be argued that these packages need more space to accommodate programs catering for different priorities and therefore have less core space available to store a stiffness matrix. This argument is partially countered in the case of Lusas in that it uses an overlay technique so that only the parts of the package needed for the particular problem whose solution is required are stored in core.

11.2.2 In relation to Sap IV, its apparent inability to handle large structures has been noted by others. Krishnamoorthy, C.S and Selvanathan, J (49) report that "In its standard form Sap IV does not have the capability of analysing large structural systems using the substructuring technique". They go on to describe their project which is designed to include this facility so that Sap IV can be used to analyse large offshore structures.

11.2.3 The problem referred to in 11.2.1 was twice redefined to simulate the increase in both the total number of degrees of freedom and the bandwidth. This redefinition was actually obtained by reducing the size of the real number array and leaving the problem undisturbed. After an initial successful run with the 12K array reduced to 11.5K, failure occurred after the reduction of the array by a further 0.5K.

11.2.3.1 Failure of the package to handle the simulated extra numbers of degrees of freedom within the same core size was due to the calls to disc made by subroutine REDUCE exceeding a preset limit. This problem could have been overcome by either of two methods. The first method is to remove the limit on the number of calls to disc. The second and better method is to decrease the size of the larger stiffness matrices of individual panels by redefining the structure represented by those panels into more panels. This failure to handle extra degrees of freedom was only precipitated by attempting to increase the structure's bandwidth. If the bandwidth had been kept down the number of degrees of freedom for the whole structure could have been increased until the City University's limit on run time or disc calls was reached, whilst still not exceeding the original array size of 12K.

11.2.4 Schrem, E (50) writes "Unfortunately, the band algorithms and wavefront methods are limited in the manageable problem size, although they could be extended to handle larger problems at the expense of reduced efficiency and simplicity". The approach used in this thesis has overcome the problems raised by Schrem and moved the barrier of size back to where problems previously considered unmanageable can now be solved without loss of accuracy or simplicity.

11.3 Final achievements in relation to 11.1.1.2.

11.3.1 In considering 11.1.1.2 full advantage has been taken of the geometric simplicity of many engineering structures to give minimum preparation of data. At the same time more complex and unrepetitive structures can be analysed. Finifter, D Wyniecki, P and DeCastel, J (51) describe the French-developed package Demain, as similarly taking advantage of the repetitive nature of some structures.

11.3.2 Appendix C contains examples of the amount of data needed for the same problem when analysed by Lusas, Sap IV and this package. A strict comparison of input data size as being a measure of the time spent on data preparation is far from correct. The data for this present work is very quickly formed. To the time element in comparing data generation of different packages must be added a component to allow for the difficulty of getting large amounts of data correct and the complication of suppressing unwanted degrees of freedom. The data for this work is very easily assembled and only freedoms which exist and which are known to have zero displacements are mentioned in the data for suppression. The numbering by hand of large numbers of nodes and elements is not required, in this work only the corners of panels need to be numbered by hand. The only node co-ordinates which need to be defined are those at the corners of panels. This task is alleviated by having a three dimensional node mesh generator requiring much easier data definition than either of the two comparison packages. Point loads, couples, thermal variations, gravity and accelerations are simply defined as they are for Lusas and Sap IV. Hydrostatically applied loads are prepared fully within the package and any orientation of the panel is coped with. This is not a feature unique to this package, but is something neither Lusasnor Sap IV are able to handle.

11.3.3 For very large geometrically repetitive structures for which this package was primarily designed, the data generating by hand is greatly reduced by the package's ability to recover and copy data applicable to a previous panel for any nominated subsequent panels.

11.3.3 An accurate figure cannot be given for a true comparison between data preparation times, but the value for time spent in data preparation for this package must be well below one tenth of the time required for either Lusas or Sap IV.

11.4 Final achievements in relation to 11.1.1.3.

11.4.1 In considering 11.1.1.3, the examples of appendix C show that there is no loss of accuracy produced by this package and that it is possible to reduce the number of degrees of freedom needed by Lusas or Sap IV and still maintain the same degree of accuracy.

11.4.2 Test 1 of appendix C shows that to achieve the degree of accuracy attained by distortion mode elements whose derivation was shown in chapter 4 with four node compatible elements requires over 10 times more elements. It is therefore possible that with the same number of degrees of freedom to attain a higher accuracy or alternatively by reducing the number of degrees of freedom have a smaller stiffness matrix whilst maintaining the standard of accuracy.

11.5 Final achievements in relation to 11.1.1.4.

11.5.1 In considering 11.1.1.4, as explained in 11.1.2, it is not possible to give definitive comparisons between the run times for the three packages. In all cases the Sap IV package was the fastest but against this, Sap IV has very poor data generating facilities and must thereby be saving time. Lusas which has better data generating capabilities than Sap IV was the slowest of the three.

11.5.2 The test referred to at 11.4.2 shows that by using the distortion mode element rather then the simple compatible element there is corresponding saving in time. Even when a 9 node isoparametric element is used there is still a loss in comparitive run time due to the extra freedoms which now require to be assembled and increased matrix to be inverted.

11.5.3 Sabir (39) gives the following formulae for the number of arithmetical operations for the inversion of a symmetrical matrix.

Fully populated	$N^3/2$
Banded	$N * n^2 / 2 + N^2 * n$

where N is the size of the matrix, and
n is the half bandwidth.

Sabir explains that the number of arithmetical operations represents the computing effort or the 'execution time' required to solve the set of simultaneous equations.

11.5.3.1 For the purposes of comparison it is now assumed that for the type of tanks and their supports used in the type of liquefied gas carriers being dealt with in this thesis, a mathematical relationship exists between the total number of degrees of freedom, the bandwidth, the number of springs representing tank supports and the number of iterations of the stiffness matrix solution.

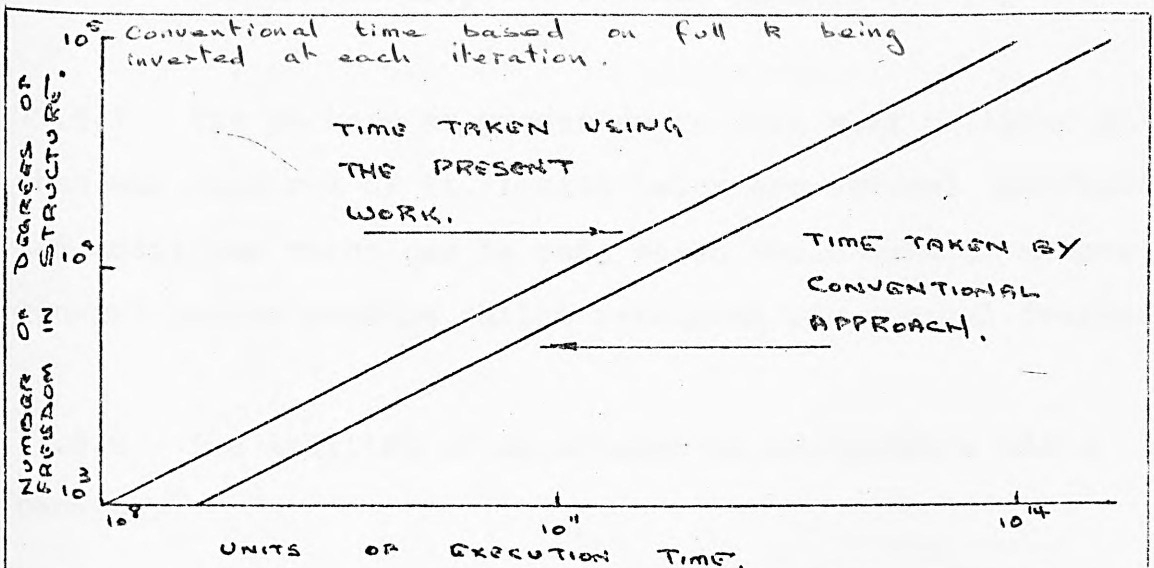
It is further assumed that this is a linear relationship and is of the form:

Bandwidth = (total number of degrees of freedom)/10

Number of springs = bandwidth/10

Number of iteration = (number of springs)/5, where half the springs remain active.

11.5.3.2 Based upon Sabir's 'execution time' formulae and the assumptions made in 11.5.3.1, Fig. 11.1 was prepared.



Comparison of execution times taken with and without using the present work for the structural analysis of a ship borne liquefied petroleum gas tank.

Fig. 11.1

11.5.4 In using the present work on the specific problem it was designed to handle, it can be seen from Fig. 11.1 that there is a considerable saving in execution time, over the problem being solved by the usual simplistic approach which requires the full matrix to be inverted each time a decision is required as to whether any more insulating supports have become unattached from the structure. From the figure it can be seen that for example a problem with 10^5 degrees of freedom will use of the order of 200 times more execution time if not run using the present work. If the existing method does not contain an inbuilt technique for the removal of the stiffness terms of the unattached supports then the total time for running an analysis not using the present work will rise considerably to take account of the human factor involved in ammending the problem for each iteration.

11.6 Further development of this package.

11.6.1 The package as presented in this work achieves all that was required of it. Listed below are several modifications and additions which can be made which would make it a more general pupose package whilst retaining its special features.

11.6.2 The addition of an element to accommodate plate bending.

11.6.2.1 Such elements for rectangular and triangular shapes are given in Przemieniecki (32). The assembly technique described in chapter 7 may still be used, though the

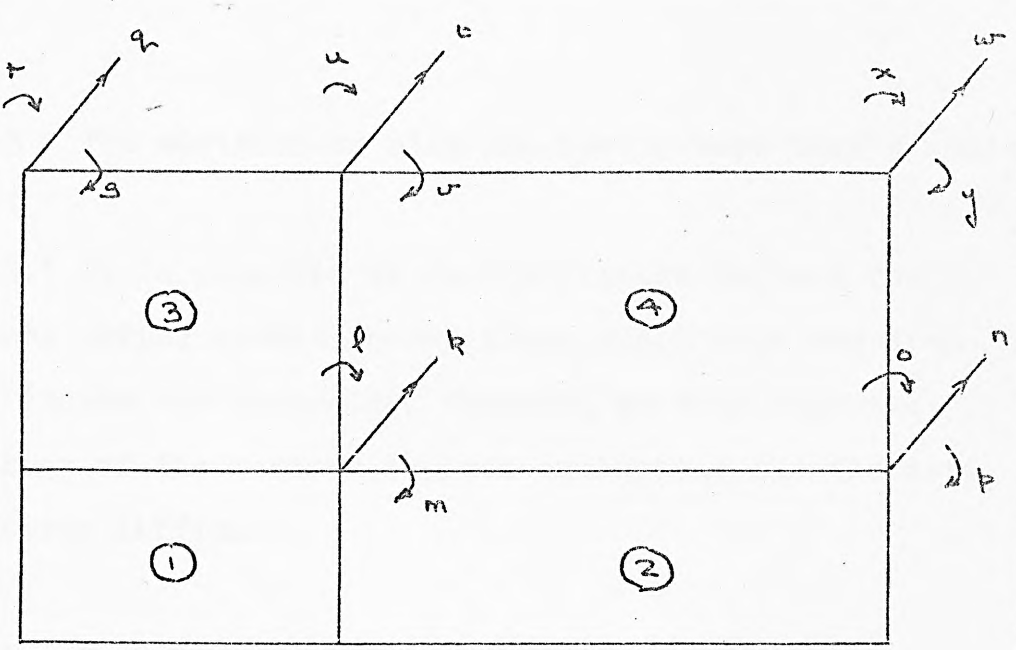
COLUMN NUMBER.

	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y
k. Row	$10,10_1$ $7,7_2$ $4,4_3$ $1,1_4$	$10,11_1$ $7,8_2$ $4,5_3$ $1,2_4$	$10,12_1$ $7,9_2$ $4,6_3$ $1,3_4$	$7,10_2$ $1,4_4$	$7,11_2$ $1,5_4$	$7,12_2$ $1,6_4$	$4,7_3$	$4,8_3$	$4,9_3$	$4\ 10_3$ $1,7_4$	$4,11_3$ $1,8_4$	$4,12_3$ $1,9_4$	$1,10_4$	$1,11_4$	$1,12_4$
l Row		$11,11_1$ $8,8_2$ $5,5_3$ $2,2_4$	$11,12_1$ $8,9_2$ $5,6_3$ $2,3_4$	$8,10_2$ $2,2_4$	$8,11_2$ $2,2_5$	$8,12_2$ $2,2_6$	$5,7_3$	$5,8_3$	$5,9_3$	$5,10_3$ $2,7_4$	$5,11_3$ $2,8_4$	$5,12_3$ $2,9_4$	$2,10_4$	$2,11_4$	$2,12_4$
m Row			$12,12_1$ $9,9_2$ $6,6_3$ $3,3_4$	$9,10_2$ $3,4_4$	$9,11_2$ $3,5_4$	$9,12_2$ $3,6_4$	$6,7_3$	$6,8_3$	$6,9_3$	$6,10_3$ $3,7_4$	$6,11_3$ $3,8_4$	$6,12_3$ $3,9_4$	$3,10_4$	$3,11_4$	$3,12_4$

Coefficient numbers of bending stiffness vector.

Fig. 11.2

number of degrees of freedom per node has increased from two to three. Fig. 11.2 shows the make up of the stiffness vector for bending of a rectangular element. The nomenclature of this figure is as that used for Fig. 7.3. To accommodate this vector an extra 42 locations must be made available to give a new vector of 61 locations.



Order of freedoms and plate numbers
as used in Fig 11.2

Fig. 11.3

Similarly a vector may be formed for a triangular bending element. The existing subroutine ASSEMB already considers the possibility of five freedoms at each node so that in that respect no basic modification to the subroutine is necessary. A change would be required to ensure that the appropriate part of the new vector from PLATEK was loaded at the respective degree of freedom. Currently this is done twice and only if there is a stiffener present at the node are any further freedoms considered. It will be necessary to change the data generating of the freedom numbers. This will actually make the programming easier since now every node will be allocated five freedom numbers, whereas at the moment a search of the stiffener distribution is necessary to ascertain how many freedoms apply at each node.

11.6.3 The addition of elements having more than 4 nodes.

11.6.3.1 It is possible to form stiffness vectors for elements having several nodes along their side and even nodes inside the perimeter. However, in this case the formation of the vectors becomes cumbersome and the data generating difficult.

11.6.4 The allowance of edge stiffeners.

11.6.4.1 In the original philosophy it was considered that welded marine structures used stiffeners at corners infrequently, and it was decided that the space that would be taken up in the program allowing for sixth order freedoms which might appear when transforming edge rotational freedoms into global axes was not justified. If at some time it was decided to put plate edge stiffeners into the package, some of the problems which could be reasonably anticipated would not arise as the program already allows a sixth location in indices. The existing package will already cope with plate edge stiffeners if it is a free edge.

11.6.5 Reducing computer time.

11.6.5.1 Facilities exist to copy input data of identical panels, thus saving the analyst preparation time. This idea could be advantageously extended to the computer processing. Input data is presently processed into the form of a reduced stiffness matrix for each panel. All the coefficients representing the internal freedoms are already stored on disc. It would seen that not much additional work would be needed to arrange to store on disc the reduced stiffness matrix, so that when a copy integer was detected the original stiffness matrix could be recovered and added to the existing stiffness matrix.

11.6.6 The copy instructions are too restrictive.

11.6.6.1 Though the system of copying data of identical panels works well, there are cases where the panels are different only because of a loading or restraint change. A second integer could be placed after the four corner numbers such that it would trigger a response causing a previous panel's data to be copied in full and processed or copied in full but then overwritten with fresh data for the loads and or the restraints.

11.6.7 Removal of input data restrictions.

11.6.7.1 Though it serves to keep panel sizes small the requirement that the edge of a panel shall, if it meets another panel meet only on an edge of that panel causes an increase in input data in some cases. This restriction could be removed by adding to the routine which allocates the global reference number to edge degrees of freedom, a loop to check if the current internal node lies on a grid line between two nodes which have been declared as corners. The indices of external degrees of freedom could then be supplemented by the values which would be allocated if such a trigger was activated. Such a trigger could also cause the ASSEMB subroutine to treat these freedoms as psuedo externals and prevent them from vanishing from the panel's active list before the whole of the panel's reduced matrix was formed.

11.6.8 Storage of bigk.

11.6.8.1 Reference was made in 11.2.3 to the moving back of the barrier of size. At this stage the barrier has not been removed. Core space is still required for bigk which is formed of all the active freedoms still existing. It seems possible to adapt a wave front technique to enable part of this stiffness matrix to be offloaded. When a new smallk is added it will, with the structures envisaged, never contribute non zero terms to all rows, therefore there will always be coefficients occupying space in the core which with a suitable housekeeping arrangement could be put off to disc to await recalling when some or all of its freedoms become currently active as opposed to conventionally active. It may be necessary in such cases to recall more than one disc record. The apparent logical inference from this reasoning is that an active wavefront could contain within itself a wavecrest at a hyper-active state.

11.7 Potential of mini and micro-computers.

11.7.1 The cost of computers is constantly decreasing. Small firms are able to purchase computers which were previously out of their financial range. The type of computer which management's budgetry control will allow the design department is probably a mini-computer. To narrow down the field of mini-computers available to those which are most likely to be suitable for structural analysis, Swanson, Ref. (52) has proposed a thirteen point selection criterion for structural analysis dedicated mini-computers. A scan of relevant papers, Refs (53, 54, and 55) suggests that though the potential is available, structural analysis packages to deal with large structures on micro-computers are not yet available. It is reported Ref. (56) that Lloyd's Register of Shipping using a desk top computer is able to perform analyses of structures of up to six hundred degrees of freedom. The desk top computer being used in this case has a capacity of 449K. Example 4 of appendix C which had nearly two and a half thousand degrees of freedom used only a total of 60K, which included the program storage. Neither of these parameters was a limiting value, the number of degrees of freedom could have been larger whilst at the same time the core size could have been reduced.

11.7.2 The use of micro-computers for structural analysis has been considered by Bennett and Goodno Ref. (57). Further to the handicaps of speed and language capability limits, they see the main restrictions coming from lack of core space. This restriction would not seen so limiting in the case of the work described in this thesis.

Appendix A.

Input Data.

A card which requires one or more continuation cards will be called one card.

The total number of cards required for any structure divided into N substructures is between N + 6 and (2 * N) + 6, inclusive.

CARD 1 - Title.

CARD 2 - This card contains eight integer and 2 real numbers as below:-

2.1 The number of panels into which the whole structure is divided. These may represent a truly flat panel or be the mean of a curved surface.

2.2 The number of springs in the whole structure. This number includes those springs representing parts of structures able to accommodate only compressive loadings, those representing anchor arrangements where there are no definite zero deflection freedoms and those representing an adjacent structure not being added in its own right.

2.3 The size of the integer array Idata. An upper bound estimate of the size of the array Idata may be obtained from the sum of the following;

30,

The number of corners in the structure,

Three times the number of springs,

Four times the number of types of stiffeners,

Twelve times the number of panels,

and the maximum value of the sum of the following applied to each panel;

The number of thicknesses,

The number of loads,

Twelve times the number of nodes on the panel,

Twice the number of stiffeners and twice the number of restrained freedoms.

2.4 The size of the real array Real. An upper bound estimate of the size of array Real which will be sufficient for most cases may be taken as twice the square of the number of perimeter freedoms for the panel having the most perimeter freedoms.

2.5 The number of corners in the whole structure. To minimize the amount of core space this value should be accurate though an estimate in excess will not cause an error.

2.6 Poisson's ratio.

2.7 Young's modulus.

2.8 The number of types of stiffeners for which inertia and other properties will be given on data cards.

2.9 The number of types of stiffeners for which properties are not known and data in the form of dimensions of the stiffeners will be given.

2.10 Either a 1 or 0 indicating whether data about co-ordinates of corners will be entered individually (Option 1) or as three overall dimensions from which a mesh will be generated. This second option allows use of the first option to follow the second if there are irregularly-positioned corners whose co-ordinates are not formed by the mesh generator.

CARD 3 - Properties of stiffeners, where cross-sectional areas, inertias about their own centroids, eccentricity of centroid of cross-sectional area to plate node and overall height of stiffener are known. Dummy stiffeners are acceptable with zero for all values except inertia which must have a small value such as the inertia of a small width of plating. If 2.8 is zero this card is left out.

CARD 4 - Details of those stiffeners for which properties must be generated. The cross-sections of such stiffeners are represented as a series of rectangles and data is entered on the card in the form

$$NR,(D_1,W_1,EC_1,D_2,W_2,EC_2, \ldots D_{NR},W_{NR},EC_{NR})H$$

repeated NS times, where

NR is the number of rectangles for that stiffener type.

NS is the number of stiffener types,

W is the width of rectangle (dimension parallel to plate surface),

D is the depth of rectangle (dimension normal to plate surface).

EC is the eccentricity of rectangle's centroid to plate - stiffener interface,

H is overall depth of stiffener.
If 2.9 is zero this card is left out.

CARD 5 - This card contains details of, or data to generate details of, the co-ordinates of all corners (see 2.1 above).

Using option 0 will allow a rectangular mesh of regular or irregular spacing to be generated from the following details:

$I, X_1, N_1, X_2N_2, \ldots X_I, N_I$

$J, Y_1, N_1, Y_2N_2, \ldots Y_J, N_J$

$K, Z_1, N_1, Z_2N_2, \ldots Z_K, N_K$

where:

I, J and K are the number of groups of equally spaced corners along the x, y and z global axes respectively.

X_i is the distance along the x axis between the end corners of the i^{th} block. Similarly for the y and z axes.

N_i is the number of equally spaced corners in the i^{th} block and includes both end corners of that block.

The numbering of corners when defined by the mesh generator cause the origin to become corner 1 and the numbering to increase sequentially along the x axis to the outer edge. Numbering is progressed in the x o y plane starting each time on the oy axis as each new plane parallel to the zox plane is considered. Not until the whole of the current xoy plane is numbered does the process move to the next xoy plane away from the origin.

Using option 1 allows the input of three dimensional co-ordinates of corners. If option 1 alone is used then the series of co-ordinates is entered and they are automatically stored on the premise that the first co-ordinate applies to the first corner and that the remainder follow in sequence.

Using both options 0 and 1 allows a regular mesh to be generated together with individually defined corners, which do not fit that mesh. When using both options, only option 0 is declared. When the mesh generation is complete a comparison is made between the number of corners generated and the number declared at item 5 on card 2. If not enough corner co-ordinates have been generated then the program goes into option 1 and searches for individually defined corners. The corner numbers which will be given to these corners will follow straight on from the last number generated by the mesh.

The above five cards refer to the whole structure. Except for the last the remainder of the cards refer to individual substructures. There are not more than two cards per substructure.

CARD 6 - This contains five integer numbers. The first four are the numbers of the corners of the substructure. The first number is taken as the local origin, the numbering is anti-clockwise round the edge so that the first and second numbers define the local X axis and the first and fourth the local Y axis. Considerable saving in input data can be made in the case of identical panels. To this end the fifth integer acts as a trigger indicating:

(a) If less than zero the data for an identical substructure has already been seen and is available in the N^{th} block of stored information when N is the fifth integer. This is the only card that needs to be read for this panel.

(b) If the fifth integer is greater than zero that this part of the structure is the first of some identical units and that the following data is to be stored.

(c) If zero that there are no other parts of the structure having this data and that the following data may be overwritten.

CARD 7 - If the last integer on the last card is negative, this card is left out. In cases where the integer is zero or positive this card then contains 14 blocks of information:

Block 1 - The number of real and integer numbers on the remainder of this card.

Block 2 - Four integers, each being 0, 1 or 2. These relate to the relative slope of adjacent flat panels.

(a) 0 indicates this is a free edge.

(b) 1 indicates that any one of the units joining along this edge is not the same plane as the current panel.

(c) 2 indicates that there is only one other panel having this edge as a common edge and that both panels are in the same plane. The presence of the integer 2 will cause no terms for the degrees of freedom along this edge to be transformed to global axes alignment. This means that two panels having, a common edge must not only lie in the same plane but have the same local orientation of local axes to qualify for a 2 at this location. If the local axes are not equally directionally aligned then they must be given the integer 1. In these cases a third degree of freedom will be introduced which will be automatically suppressed.

In all cases these numbers are assigned to edges anti-clockwise from the local origin.

Block 3 - Number of stiffeners running parallel to the local Y axis.

Block 4 Number of stiffeners running parallel to the local X axis.

Block 5 - Working from the origin the types of stiffeners running parallel to the Y axis. (The i^{th} type of stiffener is the i^{th} stiffener read in by cards 3 and 4). In many cases the same stiffener type will be repeated, in this event the input may be reduced to putting only two integers, the type number and the negative value of the number of times this type is to be repeated before the edge of the panel or another type occurs. If Block 3 is zero then Block 5 is left out.

Blo-k 6 - As Block 5 for X axis and Block 4.

Block 7 - There is no need in this package to number elements. The assembly process scans across the plate parallel to the local x axis. If it can be seen that a smaller bandwidth can be obtained by numbering parallel to the local y axis, then the local origin of this panel as defined in the data should be moved to one of the adjacent corners. The number of elements on a panel is at the discretion of the structural analyst and cannot be decided by the program. This block gives the number of plate elements between stiffeners running in the y direction and secondly the x direction. It is necessary that element widths be constant between any pair of adjacent stiffeners or an edge and the adjacent stiffener. If this cannot be achieved, dummy stiffeners must be introduced as necessary. If the element numbers are not constant right along that local direction axis (this does not necessarily imply constant element size), then the number of elements in that local direction is entered a zero.

Block 8 - If neither of the 2 figures of the last block is zero, this block is left out. For each zero in last block enter:

(a) the number of figures that follow to complete the information for this local direction, and then,

(b) the number of elements between stiffeners (or edge or dummy stiffener as applicable). Repeated numbers may be reduced to 2 figures as in Block 5. If Block 8 was 0 0 then repeat form of information for number of elements in local Y direction as in first part of this block.

Block 9 - Gives the spacing along the local X axis between stiffeners running in local Y direction. In the usual case of equally spaced stiffeners it is sufficient to use 1 - 1 and this will trigger a routine to recover the plate dimension and the number of stiffeners and produce the values the program expects from block 9. If all stiffeners are not equally spaced, block 9 then consists of one integer being the number of real numbers following the complete Block 9. The real numbers are the distance from the local Y axis to the first stiffener, if this is on the edge a zero must be entered, the following figures are distances between the stiffeners in order from local Y axis, the last figure is last stiffener to trailing edge even if it is zero. If a value is repeated the routine of using the value only once and following it by the negative of the number of times the value is to be repeated may be used. If there are no stiffeners the brief data 1 followed by the length of that side is sufficient.

Block 10 - This data gives the spacing along the local y axis between stiffeners running in the local x direction. The format of this data is as for block 9.

Block 11 - Gives plate thickness details. For this block the first digit is the number of thickness changes. This is followed by a real and an integer number pair for every thickness applicable where the integer is the number of times this thickness value applies. If the whole of the remainder is of this thickness then either the number of times this is to be applied may be entered or more simply a minus one will automatically trigger all subsequent calls for a thickness value to have this value. When the above relaxations do not apply thickness values are expected for each element row in turn, starting with the local x axis first and for each row moving from the local y axis in the x direction.

In the case of near zero thicknesses representing cut outs, to avoid badly conditioned equations, degrees of freedom within the cut out area should be listed as restrained.

Block 12 - Density of structural material of this panel including stiffeners.

Block 13 - Global accelerations of panel as a ratio with the acceleration due to gravity.

1. In the x direction
2. In the y direction
3. In the z direction.

Block 14 - 1. Temperature of panel.

2. Coefficient of linear expansion.

Block 15 - If there is no hydraulic head to be considered this block is left out, otherwise this block consists of

1. The figure minus one, to cause activation of the subroutine HYDRO.

2. Density of liquid.

3. Mean head on panel in global x direction.

4. Mean head on panel in global y direction.

5. Head of liquid in global z direction above global origin.

Block 16 - Gives details of point loads including moments being applied to the panel. If there are no point loads a zero must be entered irrespective of whether or not there is a hydraulic load applied. Point load data consists of the number of point loads followed by three figures for each point load, viz

1. The local node number,

2. The order of freedom through which the load is being applied, and

3. The value of the load.

Block 17 - Gives details of the restraints applied to this panel. The first figure is the number of restraints. This is followed by a pair of integers for each restraint. The first number of the pair being the local node number and the second the order of freedom being restrained.

Block 18 - This contains details of the springs which have an ending on this panel. The first value in the list for this block is the number of springs concerned. The remainder of the data is in groups of four figures per spring ending, viz

1. The local node number on which the spring ends.

2. The order of freedom at that node to which the spring alignes. If this spring is representing an insulating block then the spring's line of action must correspond to a translation freedom.

3. A number that is the spring's individual number allocated by hand when considering the whole problem. The same spring when being entered into the data of the panel receiving its other end must be defined by the same number. A spring representing an anchor or the stiffness of adjacent structures is given its own number but this number only appears once in the data, since the other end does not go to defined structure.

4. A number indicating which block of data on card 8 applies to this spring. In the case of anchor springs where the data is derived from the existing stiffness matrix and hence requires no input data at this stage, a zero is entered for the fourth figure of this block.

This completes the data required by card 7.

CARD 8 - This card contains information about the properties of the springs. The first number on the card is the number of values on the remainder of the card. The remainder of the values on this card are the stiffness values of the springs such that the i^{th} figure refers to the i^{th} spring. The stiffness values are derived from $A * E / L$, where

A is the cross sectional area of the spring,

E is the equivalent of Young's Modulus for the spring material in compression, and

L is the length of the spring.

CARD 9 - Gives details of the units used and the amount of output required for the displacements. These details are

1. The dimensional units used, eight character spaces are allocated to receive this word.

2. The stress units used, eight character spaces are allocated to receive this word.

3. One of the words Full, Norm or Nill. If Full is used the sidplacement of every freedom at each node is printed for each nodal row of all the panels with each node having its own numbered line. If Norm is used the displacements of all freedoms are printed out consecutively in a single block per panel with no numbering.

The use of Nill suppresses any output of displacements.

Appendix B

Fault	Cause
1	Incorrect specification of corners of panel, probably due to an error in data supplied for the previous panel causing a zero to appear as one of the corner numbers.
2	As fault 1, but instead of an erroneous zero, two of the corners have the same value.
3	Thickness reading incorrect. A zero or a negative value has been found.
4	An attempt has been made to apply a load at a point not on the plate.
5	Attempt to apply a load at a non-existent degree of freedom.
6	Attempt to restrain a point which is not on the plate.
7	Attempt to restrain a non-existent degree of freedom.
8	Attempt to attach a spring to a point

off the plate.

9	Attempt to attach a spring to a non-existent degree of freedom.
10	Contents of the integer array Idata exceeds the size of Idata.
11	Panel as specified is not in one plane.
12	Panel as specified is not rectangular.
13	In this triangular panel the angle opposite the side nominated to be the hypoteneuse is not a right angle.
14	Insufficient space left in real array REAL to store the reduced stiffness matrix of next panel.
15	The first leading diagonal term of the reduced stiffness matrix is either zero or negative.
16	A leading diagonal other than the first is zero or negative.
17	The addition of the index of the current panel will cause integer array Idata to exceed its declared space.

18 The addition of the current panel's stiffness matrix will cause the real array Real to exceed its declared space.

19 Insufficient space to store spring data.

20 Attempt to add cross term of spring has found location already occupied. Check the input data defining the springs.

21 Error in reducing technique. Check input data for this panel.

22 Insufficient space for reducing technique available core unable to store one full row of stiffness matrix.

23 Some of the coefficients stored in the core and waiting to be transferred to disc have been overwritten. Increasing the size of Real by a small amount will appear to cure this fault but the error is in the input data and will then manifest itself in another form.

24 Row of stiffness matrix after reduction has a leading diagonal term equal to or less than zero. The input data

should be checked.

25 Insufficient space available for RECALL to handle a full equation. Increase the size of array Real or re-define this panel as two, thus reducing the size of the stiffness matrix to be held.

26 Attempt to recover from disc more data than is available on permitted current record length. It would be too time wasting in calls to disc to allow this analysis to continue. The structure should be re-defined to give smaller panels.

27 Failure to find suitable degree of freedom in Maxdof list. Structure is incorrectly idealized.

28 Insufficient space in array Real to allow for the addition of the latest panel's reduced stiffness matrix to the remaining compressed stiffness matrix.

29 During subroutine COMPRS a negative value has appeared in a leading diagonal location. Check input data.

30 During the subroutine for inversion a negative has appeared in a leading diagonal location. Check input data.

Appendix C

C RESULTS.

C.1 An outline of comparisons used.

C.1.1 To prove the accuracy of this package a series of tests have been run. The results of these tests which are given in this appendix have been designed to cover the various aspects of this work.

C.1.2 Confirmation of the accuracy of these test results has been established in the majority of cases by comparison with independent calculations made on the same problem. On the question of obtaining results Bass, Hokanson and Cox (1) report that most agencies prefer the calculation method over model tests, since the former is cheaper, faster and in most cases just as reliable.

C.1.3 Clough, R. W., (58) writes, "It is important to express my concern over the tendency for users of the finite element method to become increasingly impressed by the sheer power of the computer procedure and decreasingly concerned with relating the computer output to the expected behaviour of the real structure under investigation". This concern is similar to that expressed by Oden and Bathe (59). They describe the complacency and overconfidence of number-crunching experts. It is difficult to obtain the full structure details and results of strain gauges readings for tests made on large structures. To overcome this and compare the package's workings against stress outputs obtained from strain gauge readings a stiffened box section beam located in the Aeronautics Department of City University was represented using nearly a thousand nodes. The package's output compared with the previously obtained stress results is contained in this appendix.

C.1.4 When independent calculations are required, these are in most cases provided by runs made on the City University's Honeywell computer using either the Lusas package (18) or the Sap IV (19) package. In some of the simple tests, comparisons have been made with elementary bending theory. In the case of stresses resulting from a hydrostatic head comparison has been made with a run using Nastran (60). This run was one of only two which did not use City University's computer facilities. The other independent run was the principal run made by Tarpy, J. S. (61) to support his Ph.D. thesis "A technique for the finite element analysis of large complex structural systems".

C.2 List of the test undertaken.

C.2.1 Point loaded plate beam.

C.2.2 Panels connected by members transmitting compression only.

C.2.3 Hydrostatic loading. In this structural analysis of a bulkhead subjected to a hydrostatic head,the output derived from the current package is compared with that for the same problem using Nastran.

C.2.4 Stuctural analysis of a box with centre line division using SAP IV and the current package. The box is subjected to point, thermal and acceleration loads.

C.2.5 Structural analysis of a 6 bay box beam using Lusas and the current package. This representation was the largest that could be processed by Lusas within the core space made available by the City University Computer Unit.

C.2.6 Stuctural analysis of a 12 bay box beam using strain gauges and the current package.

C.2.7 Lusas run to verify features of run C.2.6.

C.2.8 Comparison with the Tarpy output (61).

C.2.9 Analysis of a liquefied gas filled tank and the surrounding structure of the ship's hull, including a small overlap with the adjacent holds.

C.3 Test 1.

C.3.1 Objective.

C.3.1.1 Comparing the convergence rates of

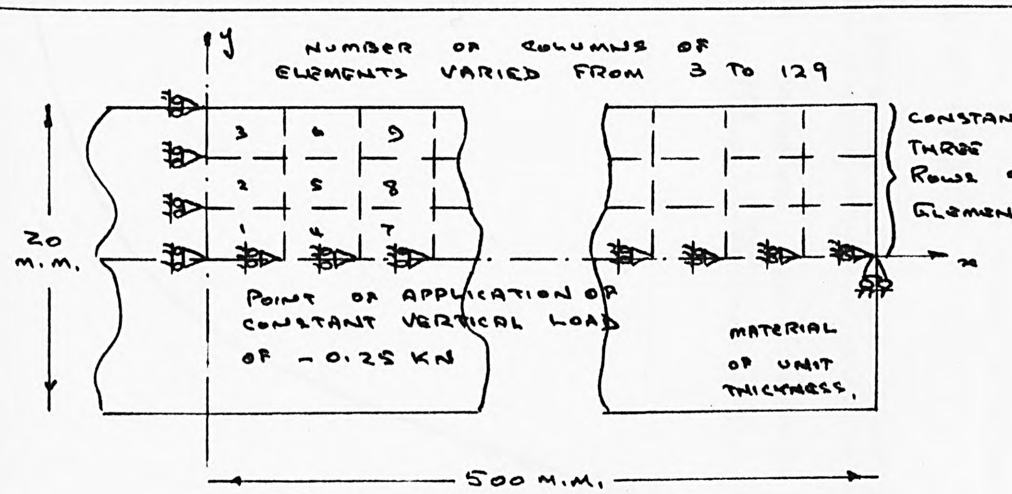
(a) Straight sided compatible four node rectangular elements.

(b) Distortion mode four node rectangular elements.

(c) Isoparametric nine node rectangular elements.

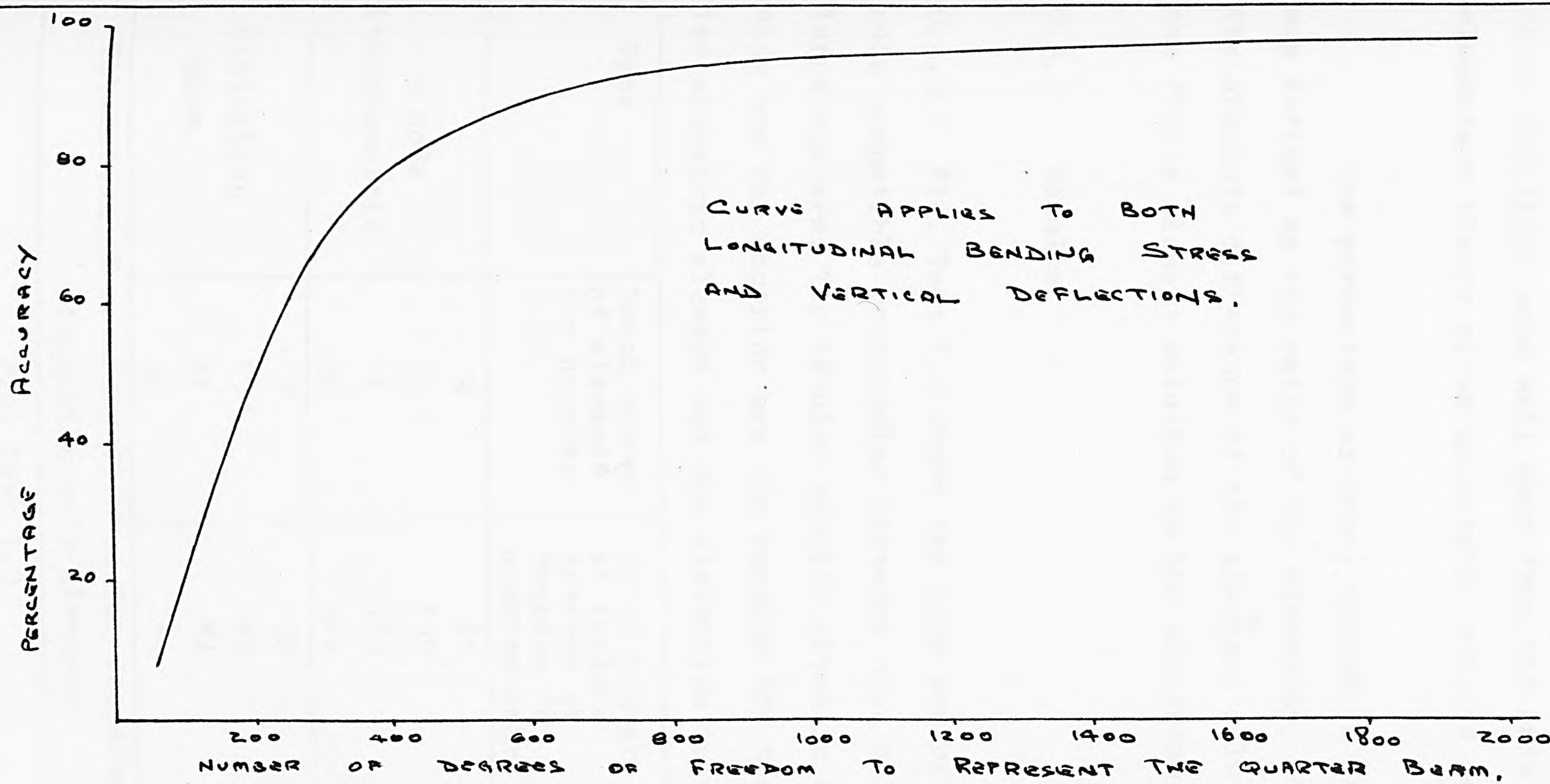
C.3.2 Description.

C.3.2.1 The model used for this comparison was a symmetrical quarter of a centrally loaded plate beam freely supported at its ends as shown in Fig. Test 1.1. The quarter beam was divided into a constant three rows of elements with the number of elements being changed by varying the number of columns.



Idealization of beam.

Fig. Test 1.1



Accuracy of four node compatible rectangular elements plotted against the idealization's number of degrees of freedom for a plate beam.

Fig. Test 1.2

C.3.2.2 The positions examined for comparisons were the centre of the quarter beam for the stress value and the centre of the full beam for the deflections. These positions were well away from the ends and therefore elementary theory gives acceptable results for comparison.

The percentage accuracy quoted in the results was defined as the ratio of the elementary solution minus the absolute difference of the elementary solution and the finite element solution to the elementary solution.

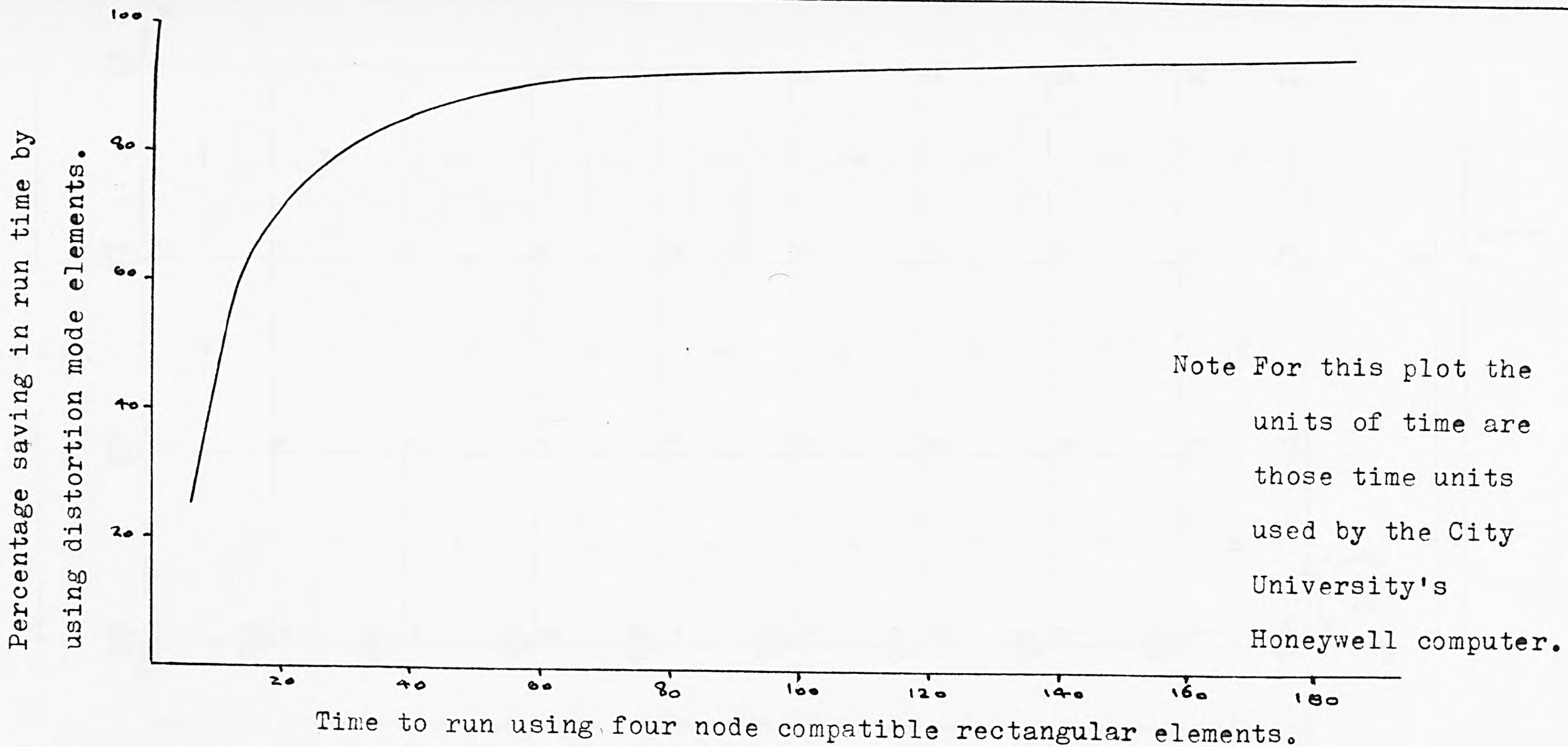
C.3.3 Results.

C.3.3.1 Fig. Test 1.2 shows the poor performance of four node compatible rectangular elements when not used in large numbers. The tabular results given in Fig. Test 1.3 show how far superior are the results for the nine node isoparametric element and the distortion mode rectangular.

Type	Total number of elements for quarter beam.	Total number of inplane degrees of freedom for quarter beam.	Percentage accuracy based on stress comparisons.
9 node isoparametric	9	81	99.97
	15	135	99.99
	21	189	99.99
	36	324	100.00
Distortion mode.	9	36	99.94
	15	45	99.89
	21	84	99.90
	36	144	99.99

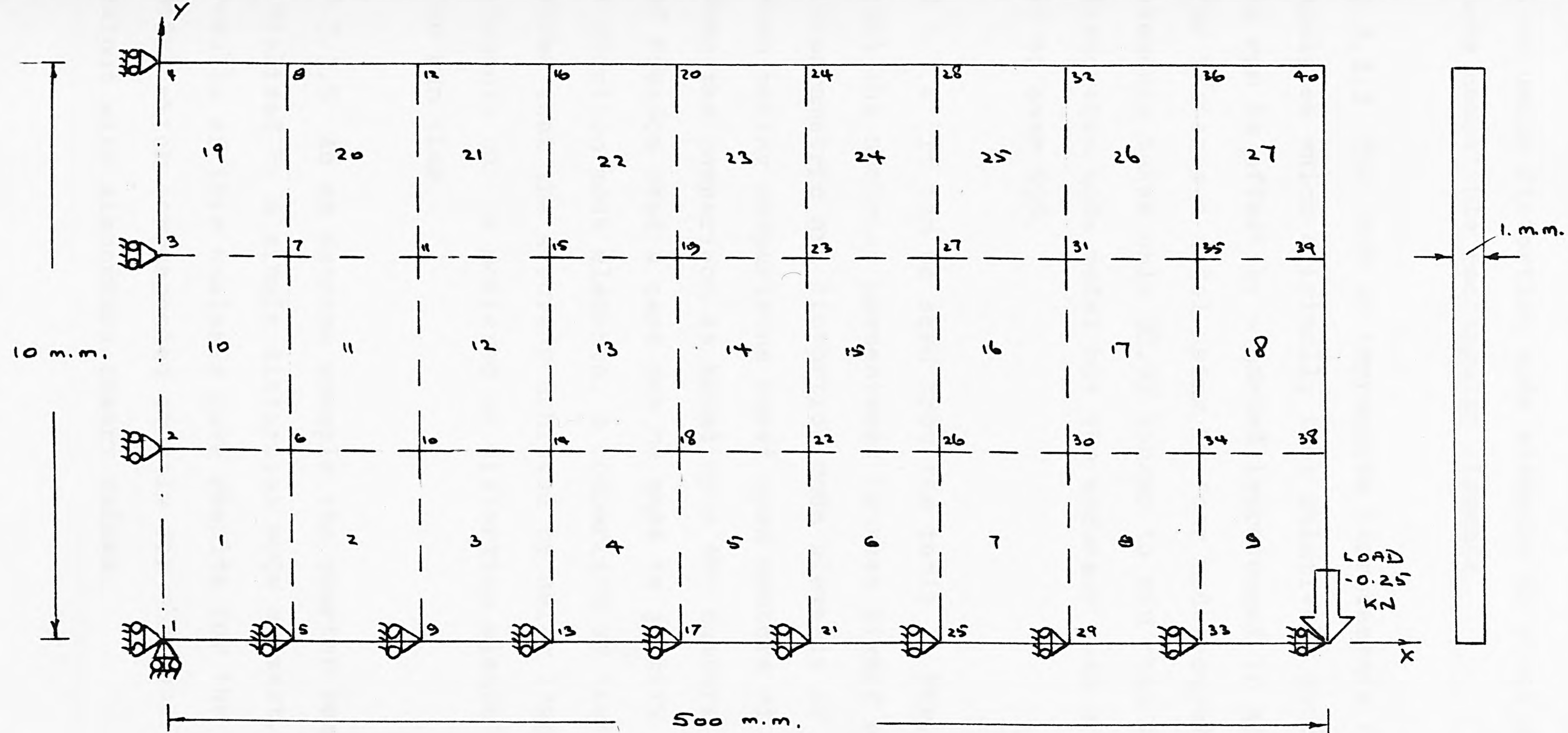
Tabulation of accuracy figures for isoparametric and distortion mode elements.

Fig. Test 1.3



Plot showing percentages of time saved , achieved by using distortion mode elements as opposed to low order compatible elements.

Fig. Test 1.4



Quarter beam represented by the data in

Fig's Test 1.7 and Test 1.8.

Fig. Test 1.5

C.3.3.2 Fig. Test 1.4 shows the percentage saving in time using distortion mode elements in place of four node compatible rectangular elements.

C.3.3.3 The lack of impressive improvements for analyses which originally took relatively short times to run is offset by a marked improvement in accuracy. For instance a model using 15 four node compatible elements takes only 38.9% longer to run than a 9 element distortion mode model but the accuracy goes up from below 10 to over 99%.

C.3.3.4 It can be seen from the table in Fig. Test 1.3 that the accuracy percentages between higher order isoparametric and distortion mode elements is very marginal when making comparisons based upon numbers of elements used. When the comparison is based upon the numbers of degrees of freedom used a case can be made to justify using distortion mode elements. A comparison of times taken shows that the accuracy achieved by using isoparametric elements can be achieved by distortion elements in half the run time.

C.3.3.5 As an extreme example the quarter beam was idealized by a single distortion mode element. The stress results of this analysis gave results for the centre of model which corresponded exactly for the four figure output with elementary theory values.

C.3.4 Input data.

c.3.4.1 Typical input data for Test1 used to obtain the results shown in C.3.3 is given in following three figures.

```
9*3 Quarter beam representation.
1 0 3000 12000 4 .3 207 0 0 1
0 0 0 500 0 0 500 10 0 0 10 0
1 2 3 4 0
54 0 0 0 0 0 0 9 3 1 500. 1 10. 1 1 -1 0 0 0 0 0 0
1 1 1 -.25 14 1 2 10 1 10 2
2 2 3 2 4 2 5 2 6 2 7 2 8 2 9 2
11 2 21 2 31 2
MM      KN/SQ.MMFULL
```

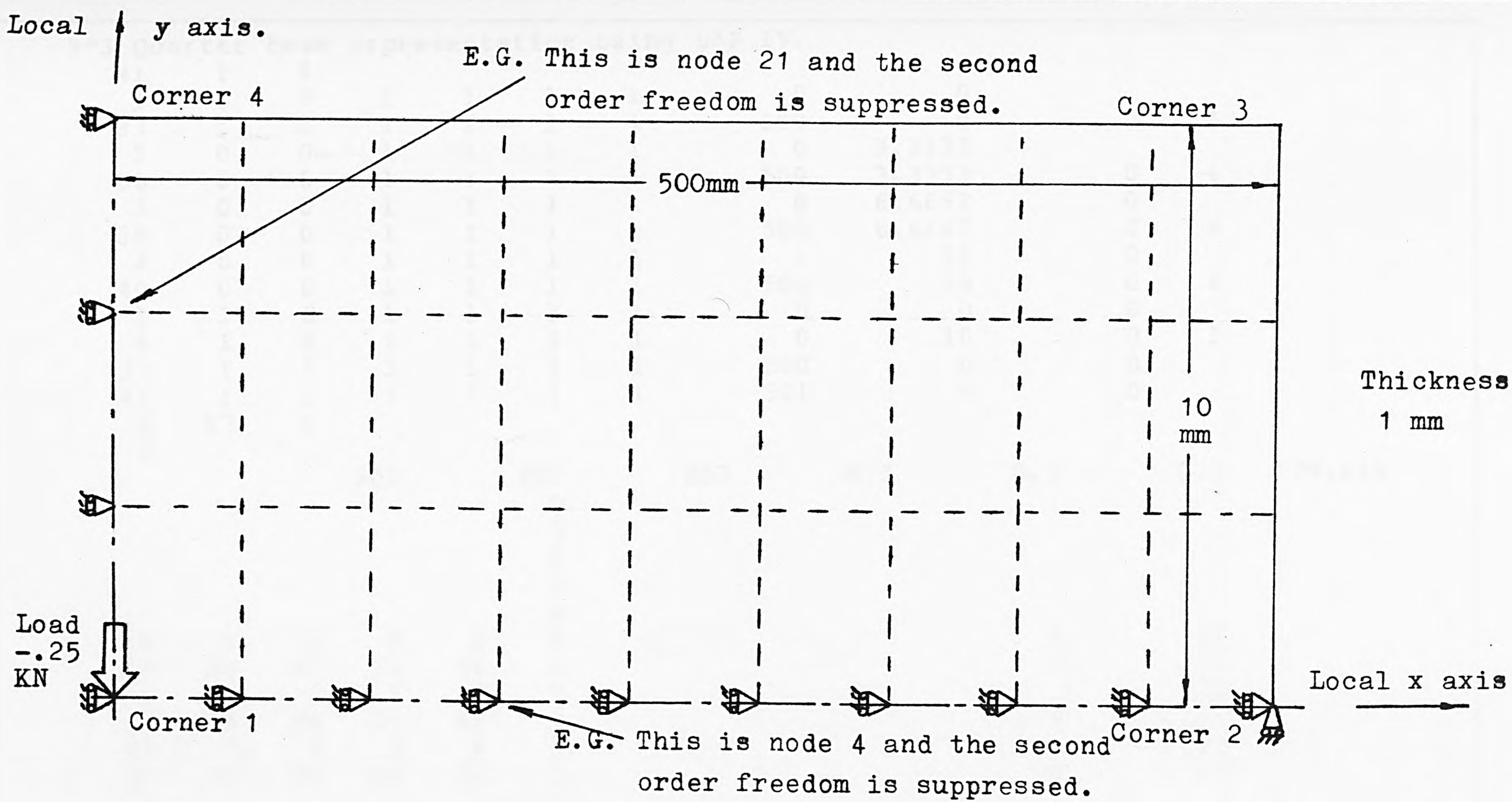
Data to run present package for the beam shown in Fig. Test 1.6 a

Fig. Test 1.6

```
PROBLEM 9*3 QUARTER REPRESENTATION OF A SIMPLE
BEAM USING THE QPM9,NINE NODE ISOPARAMETRIC ELEMENT.
UNITS KN MM
QPM9 ELEMENT TOPOLOGY
FIRST 1 1 8 15 16 17 10 3 2 9
INC   1 2 2 2   2  2  2 2 2 2 3
INC   3 14 14 14 14 14 14 14 14 14 9
NODE COORDINATES
FIRST 1 0 0
INC   1 0 10/6 7
INC 7 500/18 0 19
QPM9 GEOMETRIC PROPERTIES
1 27 1 10 10 10 10 10 10 10 10 10
MATERIAL PROPERTIES
1 27 1 207 0.3
SUPPORT NODES
1 7 1 R F
8 120 7 R F
127 0 0 R R
LOAD CASES
CONCENTRATED LOAD
1 0 0 0 -.25
END
```

Data to run the Lusas package for the beam shown in Fig. Test 1.5

Fig. Test 1.7



Quarter beam represented by the data in Fig. Test 1.6. Illustrating restraint numbering.

Fig Test 1.6a

```
9*3 Quarter beam representation using SAP IV.
  41   1   1
   1   1   0   1   1   1   1      0        0
  37   1   0   1   1   1   1    500        0     0   4
   2   0   0   1   1   1   1      0   3.3333
  38   0   0   1   1   1   1    500   3.3333     0   4
   3   0   0   1   1   1   1      0   6.6667     0
  39   0   0   1   1   1   1    500   6.6667     0   4
   4   0   0   1   1   1   1      0       10     0
  40   0   0   1   1   1   1    500       10     0   4
   1   1   0   1   1   1   1      0        0     0
   4   1   0   1   1   1   1      0       10     0   1
  37   1   1   1   1   1   1    500        0     0
  41   1   1   1   1   1   1    501        0     0
   3  27   1
   1
          207         207        207      0.3       0.3       0.3    79.615
                        0
                        0
                        0
                        0
                        0
   1   1   5   6   2    1                            0    10
   9  33  37  38  34    1                            4    10
  10   2   6   7   3    1                                 10
  18  34  38  39  35    1                            4    10
  19   3   7   8   4    1                                 10
  27  35  39  40  36    1                            4    10
   1   1           -0.25
       1       1        1    1
                        0
                        0
```

Data for SAP IV for the analysis of the beam shown in Fig. Test 1.5
Fig. Test 1.8

C.3.3.6 Of other comparison runs made but not reported here, were direct tensile, direct compression and shear cases. All of which showed that the distortion element acted as well in those modes as in bending. Additional tests were considered necessary since it might be thought that the simple bending test favoured the distortion mode element. Study of the formulation of the distortion mode element shows that equal weight is given to the distortion due to straight pulls and shears.

C.4 Test 2.

C.4.1 Objective.

C.4.1.1 This test is not intended to be a serious analysis but rather an example of the package's handling of spring systems.

C.4.1.2 To compare the results obtained when using the present work with those obtained using elementary bending theory for two cantilever beams connected by springs or struts unable to transmit forces which would put them into tension.

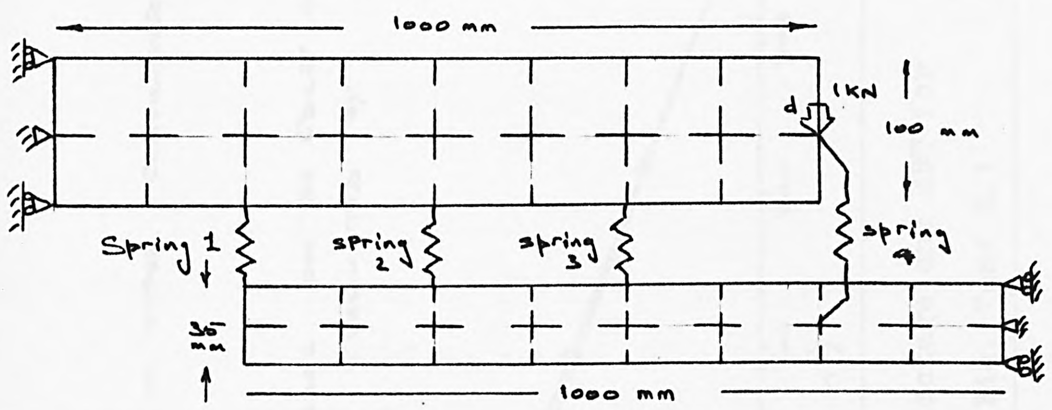
C.4.1.3 To compare the results obtained in C.4.1.2 with those obtained when using restraints on one beam which are springs having either analyst defined or self defined stiffness properties.

C.4.2 Description.

C.4.2.1 The model used for this analysis was two cantilevers as shown in Fig. Test 2.1. For the analysis a point load was applied vertically downwards at point d. The load value was 1 KN. Both beams were considered to have unity thickness with Young's modulus and Poisson's ratio set at 207 KN/mm^2 and 0.3333 respectively. Springs 1,2 and 3 were given stiffnesses of 10 KN/mm, whereas spring number 4 was given a stiffness of 55 KN/mm.

C.4.3 Results.

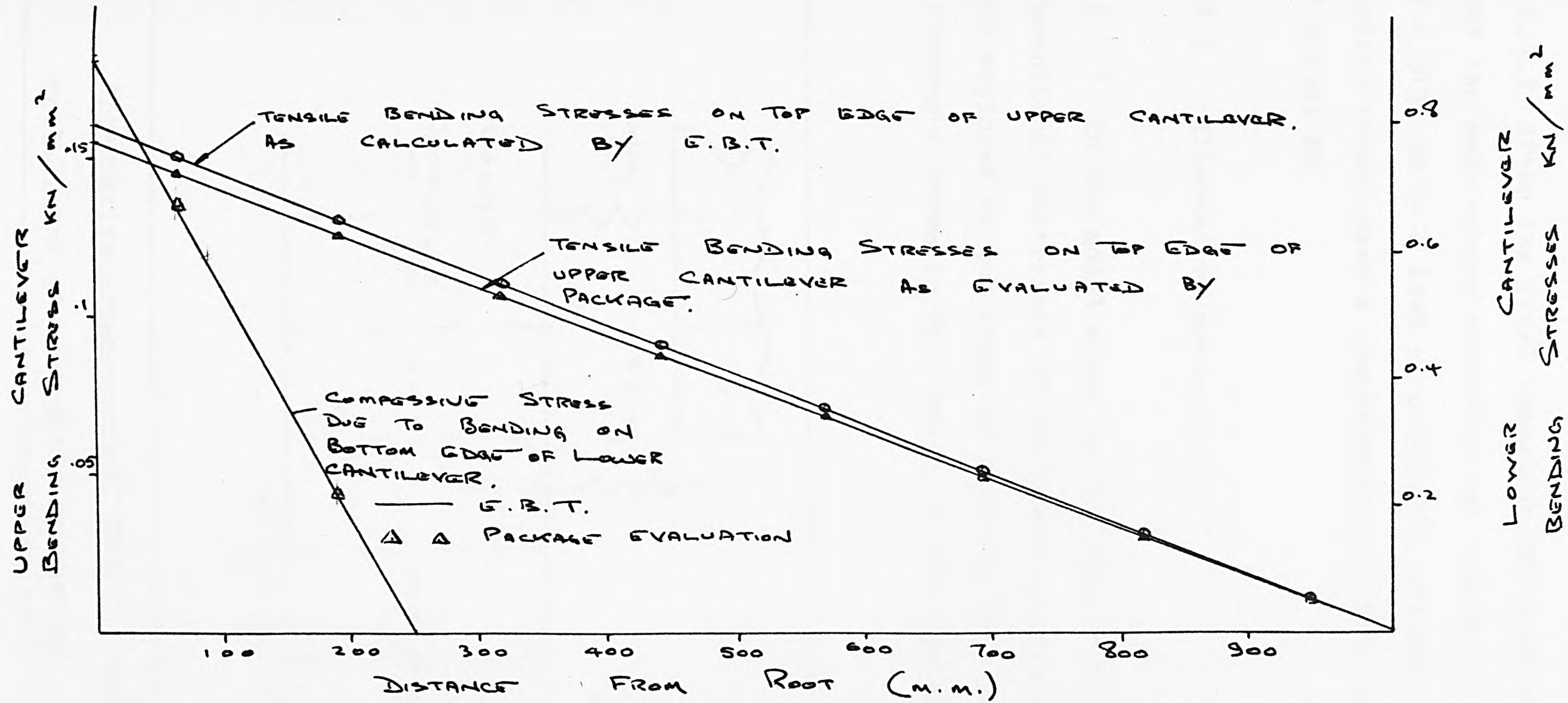
C.4.3.1 A plot of the stresses along one edge of the beam is shown in Fig.Test 2.2. This figure also contains the stress plot obtained using elementary bending theory and shows excellent correlation.



Idealization of two cantilever beams connected by comression only springs.

Fig. Test 2.1

C.4.3.2 The matrix for final inversion contains only eight rows of coefficients corresponding to the ends of the four springs. After the first inversion, two springs, numbered 1 and 3 were found by the package to be in tension and their contributions to the final stiffness matrix were deleted. After the second inversion, spring number 2 was found to be in tension and this was deleted. After the third inversion, the only remaining spring, number 4 was found to be in compression as was expected. From this stage the package proceeded and printed out the displacements of all freedoms and the stresses at element centroids.



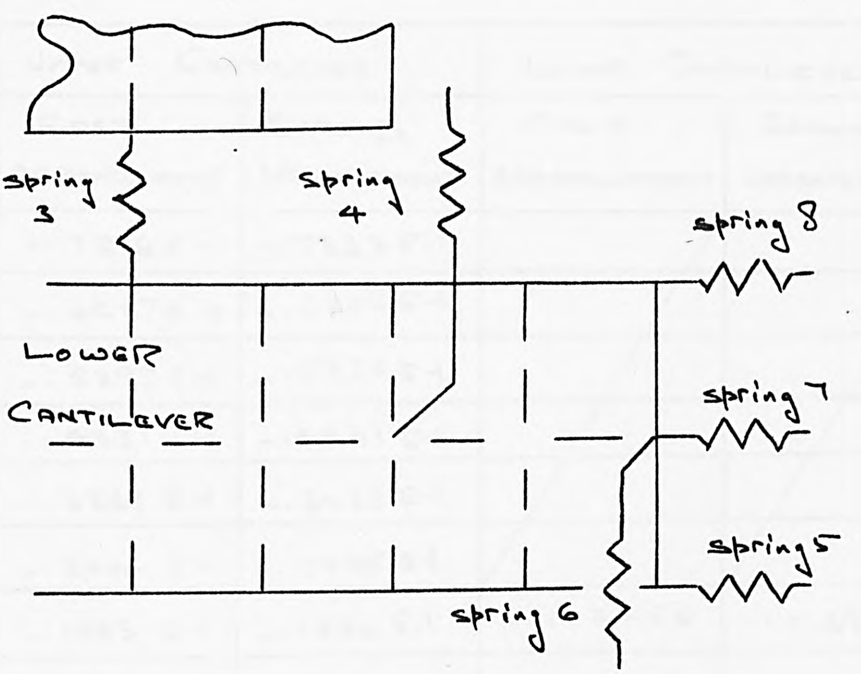
Plot of stresses found from the current package and EBT for the beam and spring arrangement shown in Fig. Test 2.1

Fig. Test 2.2

C.4.3.3 After the final inversion the analysis showed that the only spring operating was compressed a distance of 0.0135 mm by a load of 0.7433 KN, whereas engineer's bending theory gave a compression of 0.0133 mm by a load of 0.7324 KN.

C.4.4 Alternative arrangements.

C.4.4.1 In the model shown in Fig. Test 2.1, the conventional restraints of the lower cantilever have been replaced by two types of springs. This gives the arrangement shown in Fig. Test 2.3. The springs numbered



Alternative arrangement of Test 2, showing replacement of restraints by springs.

Fig. Test 2.3

5,7 and 8 have had their stiffness value defined in the input data. Spring number 6 has had its stiffness value calculated by the program instructions. For each of these four springs only the one end attached to the structure is defined.

C.4.5 Results applicable to the alternative arrangements.

C.4.5.1 Tabular comparisons of bending stresses at the elements' centroids for the bottom row of elements in the two arrangements are given in Fig. Test 2.4. Elements are numbered along the beams from the bottom left hand corner.

ELEMENT NUMBER	UPPER CANTILEVER		LOWER CANTILEVER	
	FIRST ARRANGEMENT	SECOND ARRANGEMENT	FIRST ARRANGEMENT	SECOND ARRANGEMENT
1	-.7216E-1	-.7333E-1		
2	-.6257E-1	-.6358 E-1		
3	-.5293E-1	-.5379 E-1		
4	-.4331 E-1	-.4401 E-1		
5	-.3369 E-1	-.3423 E-1		
6	-.2406 E-1	-.2445 E-1		
7	-.1443 E-1	-.1466 E-1	-.1134 E0	-.1128 E0
8	-.4839 E-2	-.4917 E-2	-.3419 E0	-.3400 E0

Comparison of centroid stresses obtained when using fixed restraints (1^{st} arrangement) and spring restraints

Fig. Test 2.4

C.4.6 Input data

C.4.6.1 The Fig's Test 2.5 and Test 2.6 list the data required to analayse the structures shown in Fig's Test 2.1 and Test 2.3 respectively.

```
SPRING CONNECTED CANTILEVERS BOTH
HAVING CONVENTIONAL RESTRAINTS.

2 4 3000 12000 8 .3333 207 0 0 1
0 0 125. 1000. 0 125. 1000. 0 225. 0 0 225.
250. 0 80 1250. 0 80 1250. 0 115. 250. 0 115.
5 6 7 8 0
48 0 0 0 0 0 0 8 2 1 1000 1 35 1 1. -1 0 0 0 0 0 0 0
4 9 2 18 1 18 2 27 2
4 19 1 1 1 21 1 2 1 23 1 3 1 25 1 4 2
1 2 3 4 0
51 0 0 0 0 0 0 8 2 1 1000 1 100 1 1 -1 0 0 0 0 0 0
1 18 1 -1.0
4 1 2 10 1 10 2 19 2 4 3 1 1 1 5 1 2 1 7 1 3 1 9 1 4 2
2 10. 55.0
MM        KN/SQ.MMFULL
```

Listing of data used by the present package to analyse the two beams shown in Fig. Test 2.1

Fig. Test 2.5

```
SPRING CONNECTED CANTILEVERS ONE HAVING
ITS RESTRAINTS FORMED BY SPRINGS.

2 8 3000 12000 8 .3333 207 0 0 1
0 0 125. 1000. 0 125. 1000. 0 225. 0 0 225.
250. 0 80 1250. 0 80 1250. 0 115. 250. 0 115.
5 6 7 8 0
56 0 0 0 0 0 0 8 2 1 1000 1 35 1 1. -1 0 0 0 0 0 0 0
0 8 9 2 5 0 18 1 6 0 18 2 7 0 27 2 8 0
19 1 1 1 21 1 2 1 23 1 3 1 25 1 4 2
1 2 3 4 0
51 0 0 0 0 0 0 8 2 1 1000 1 100 1 1 -1 0 0 0 0 0 0
1 18 1 -1.0
4 1 2 10 1 10 2 19 2 4 3 1 1 1 5 1 2 1 7 1 3 1 9 1 4 2
2 10. 55.0
MM        KN/SQ.MMFULL
```

Data required by present package to analyse the two beams shown in Fig. Test 2.3, where the resraints of one beam have been replaced by springs.

Fig. Test 2.6

C.4.7 Comments on test 2.

C.4.7.1 In addition to running the test with four springs as shown in Fig. Test 2.1 the test was also run defining all the springs as being attached to internal nodes and in another case only the one spring which the first run had shown would remain effective in the structure, was defined. Identical results were obtained for all these runs. The significance of this result is that in forming the final stiffness matrix in the case with four springs, as the assembly routine detects a spring end located on a degree of freedom at an internal node, that degree of freedom is subsequently treated as an external degree of freedom. This means that different sets of equations are set up depending on whether or not a spring end is on the perimeter of the panel. The results are the same even though different assembly routines were used. In the case using only one spring the final stiffness matrix for inversion was only 2 x 2 but since identical results were obtained, the technique used for removing the contribution of springs which were in a state of tension was satisfactory.

C.4.7.2 Examination of Fig. Test 2.2 giving the plot of stresses from this analysis shows that there is a difference between the package evaluated stresses and the values expected as predicted by simple bending theory. This is because of the nature of the distortion mode element to give accurate stresses when compared with simple bending as already shown in Test 1, but giving slight errors in displacement. When evaluating stresses in the beam by simple theory using the loads found from the package evaluated displacements there was total agreement with the stresses provided from the package's output.

C.5 Test 3.

C.5.1 Objectives.

C.5.1.1 To prove the functioning of the hydrostatic load generating subroutine by comparison of the bending stresses induced in the stiffeners with those from a run using Nastran.

C.5.1.2 This test run also provides an opportunity to use a triangular panel, a rectangular cut out representing the doorway and springs representing the stiffness of girders taking the bending moment at the ends of some bulkhead stiffeners.

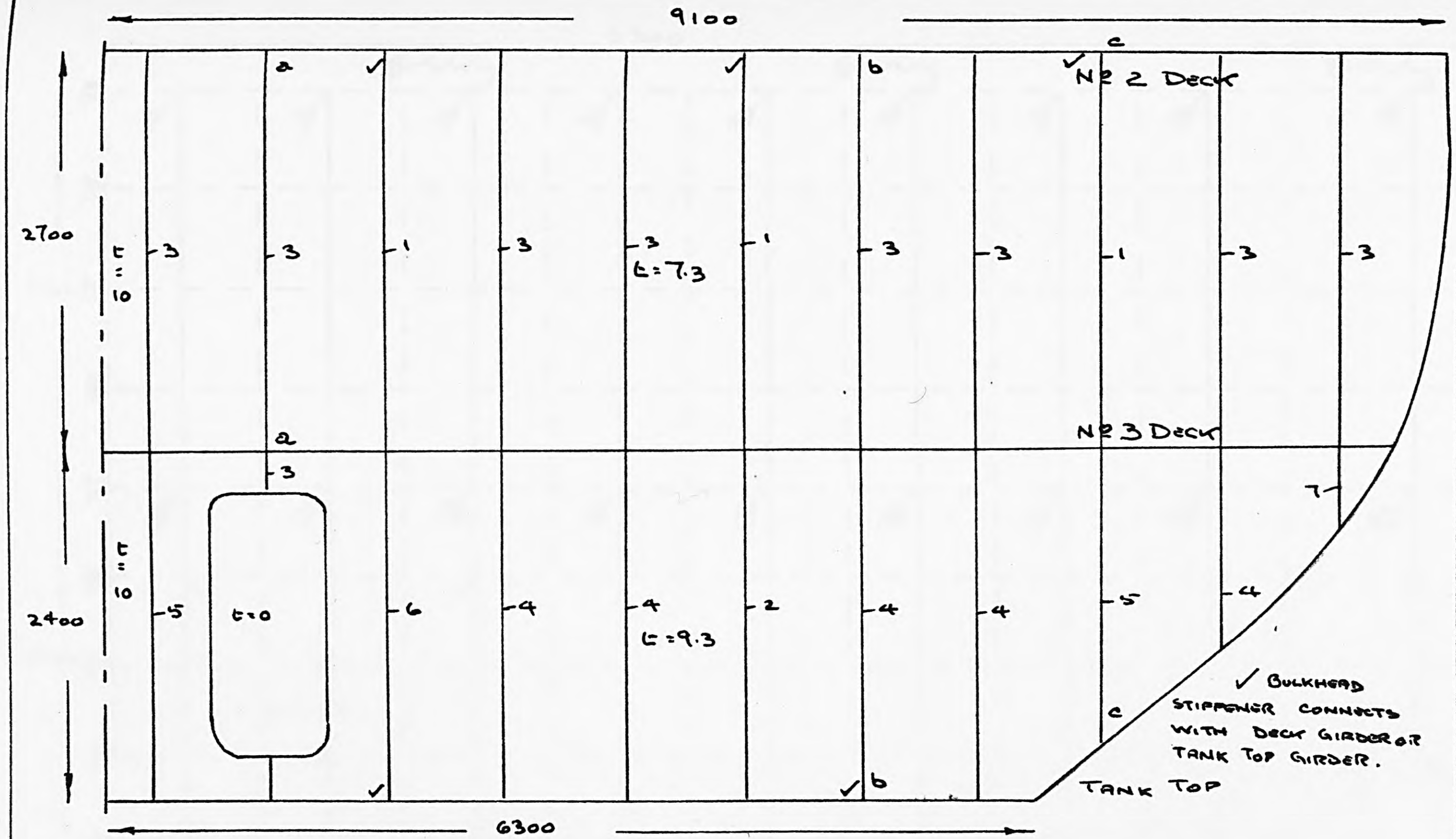
C.5.2 Description.

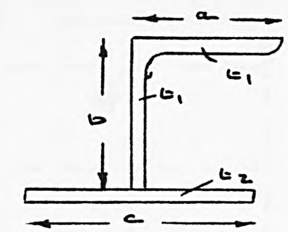
C.5.2.1 The subject of this analysis is a steel bulkhead as shown in Fig. Test 3.1. The bulkhead is subjected to a head of sea water 0.91m above the top of the bulkhead.

C.5.2.2 The stresses developed in three stiffeners are examined. These stiffeners are:

(a) The stiffener vertically above the doorway.

(b) A stiffener reaching the maximum height from number 2 deck to the tank top. The rotational freedom at the bottom of this stiffener is considered to be attached to a flexible foundation.



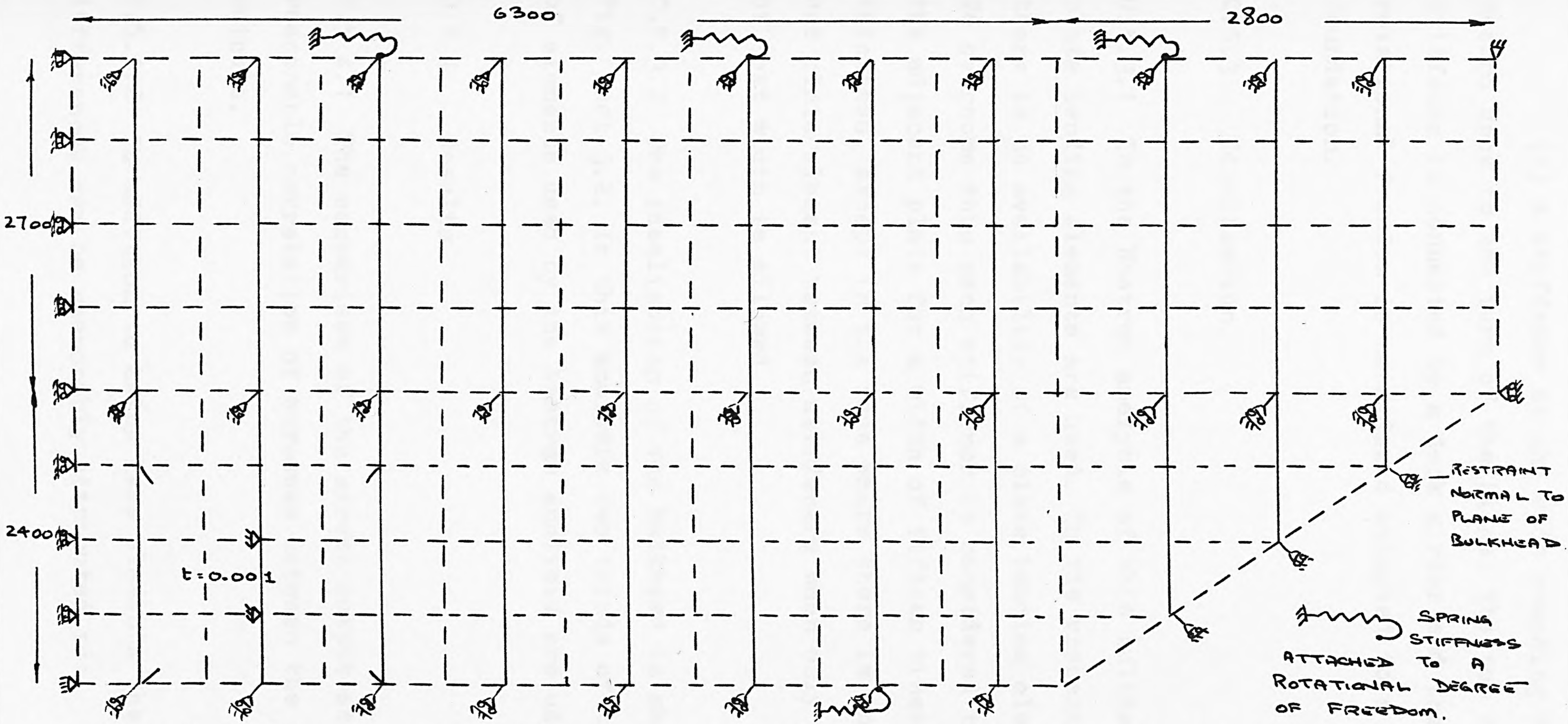


	a	b	c	t_1	t_2
1	75	75	110	10	7.3
2	90	90	140	10	9.3
3	65	75	110	8	7.3
4	75	90	140	8	9.3
5	65	75	55	8	7.3
6	0	200	140	20	9.3
7	Dummy.				

Arrangement of the steel bulkhead used in analysis Test 3.

Showing plate thicknesses and stiffener dimensions.

Fig. Test 3.1



LINES IN FULL REPRESENT BEAM ELEMENTS. DASHED LINES REPRESENT EDGES OF PLATE ELEMENTS.

Idealization of bulkhead shown in Fig. Test 3.1

Fig. Test 3.2

(c) A stiffener at the side reaching from the second deck to the turn of the bilge. The top of the stiffener is connected to a deck girder and the rotational freedom is considered attached to a flexible foundation.

C.5.3 Idealization.

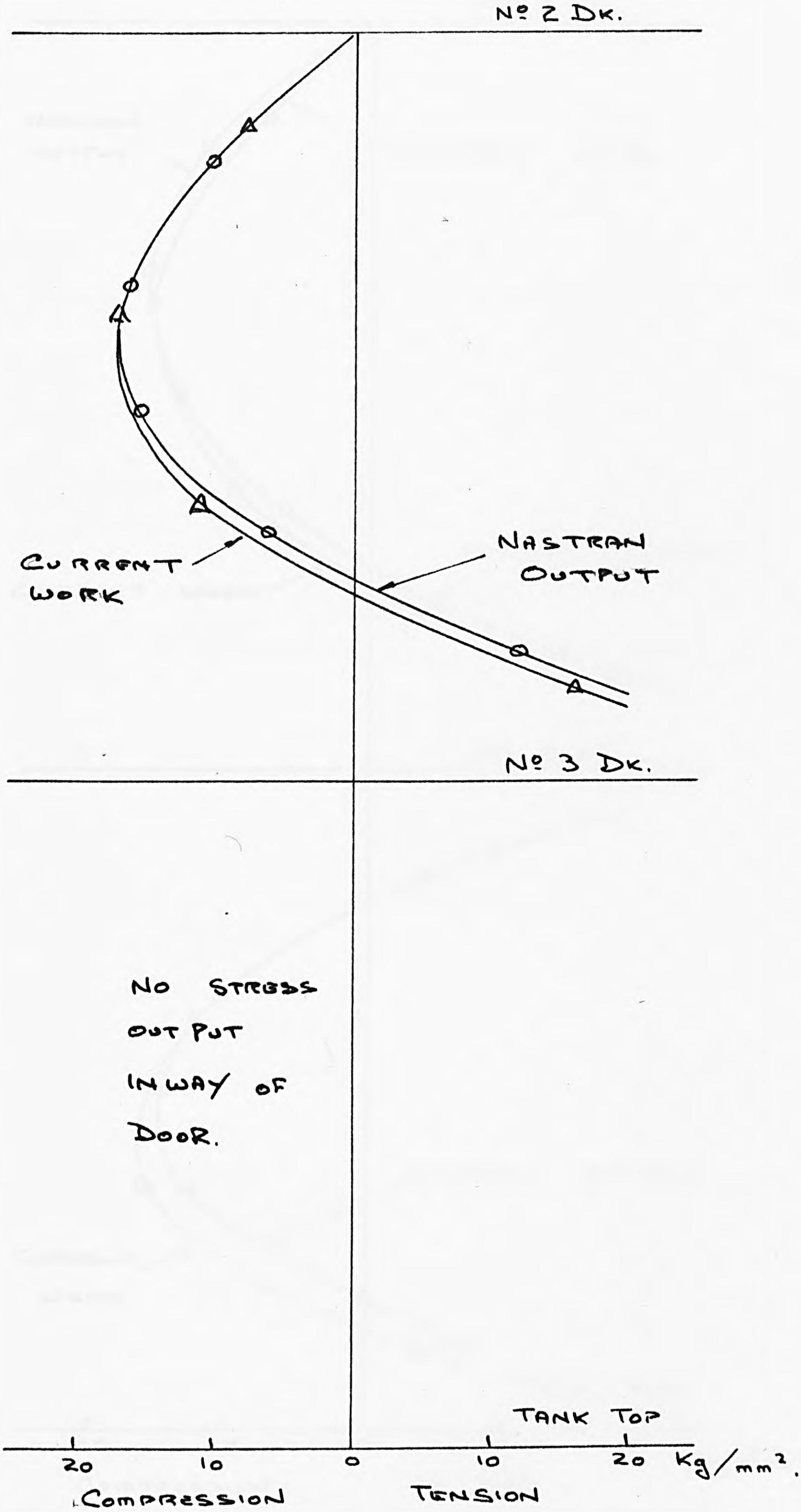
C.5.3.1 In the Nastran analysis of this bulkhead plate bending elements are used. In the present work there is no availability of a plate bending element. To overcome this each stiffener is considered to include the adjacent plate for a width of fifteen times its thickness, except in the case where there is only one plate element between stiffeners when only half of that width is allowed.

C.5.3.2 The idealization of the bulkhead is shown in Fig. Test 3.2. In this analysis two thirds of the number of elements used by the Nastran analysis are used.

C.5.4 Results.

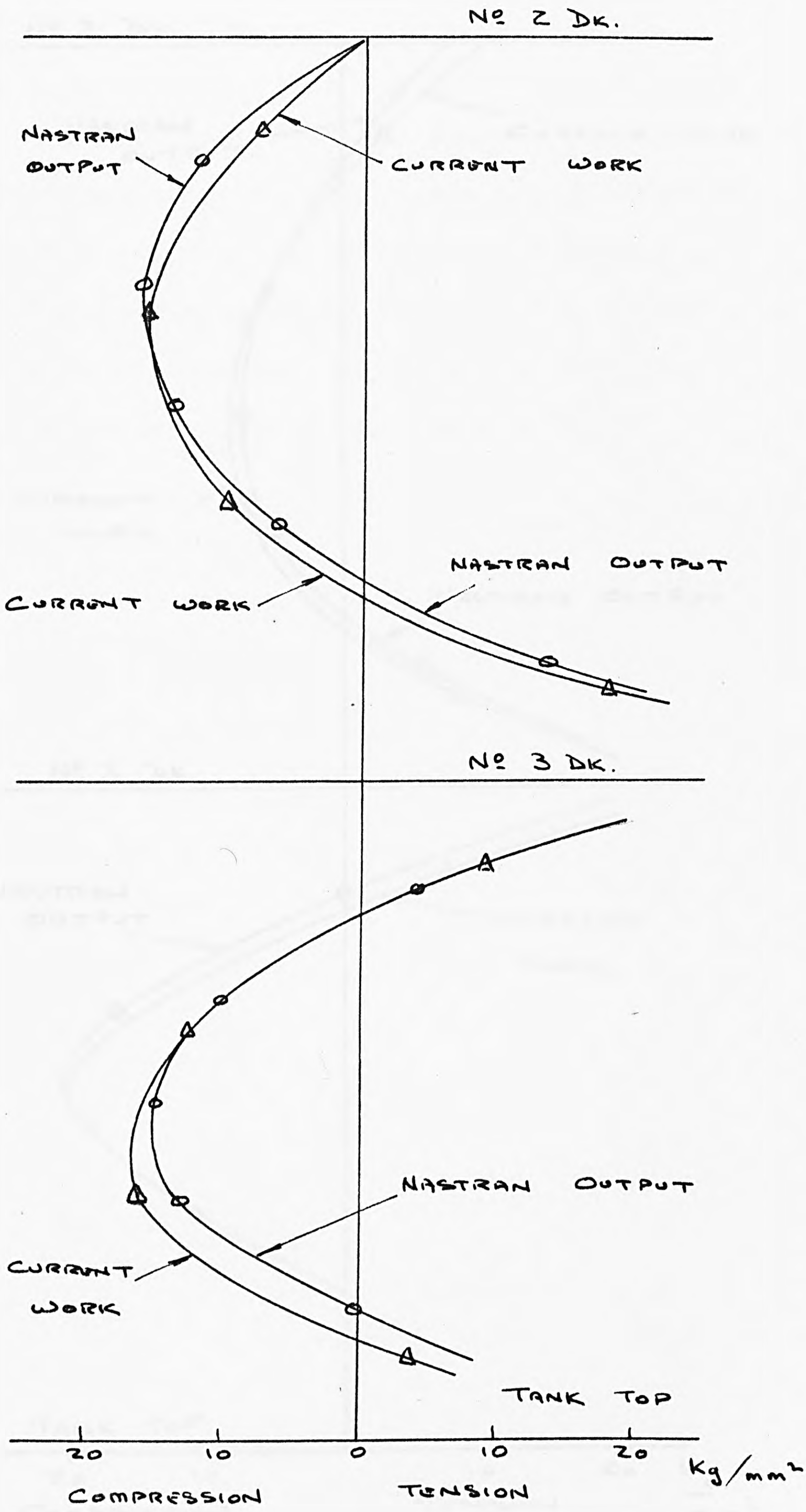
C.5 4.1 The comparison of the stress output shows reasonable correlation of stresses between the two outputs.

C.5.4.2 In addition to using less elements the small differences can be reasonably attributed to:



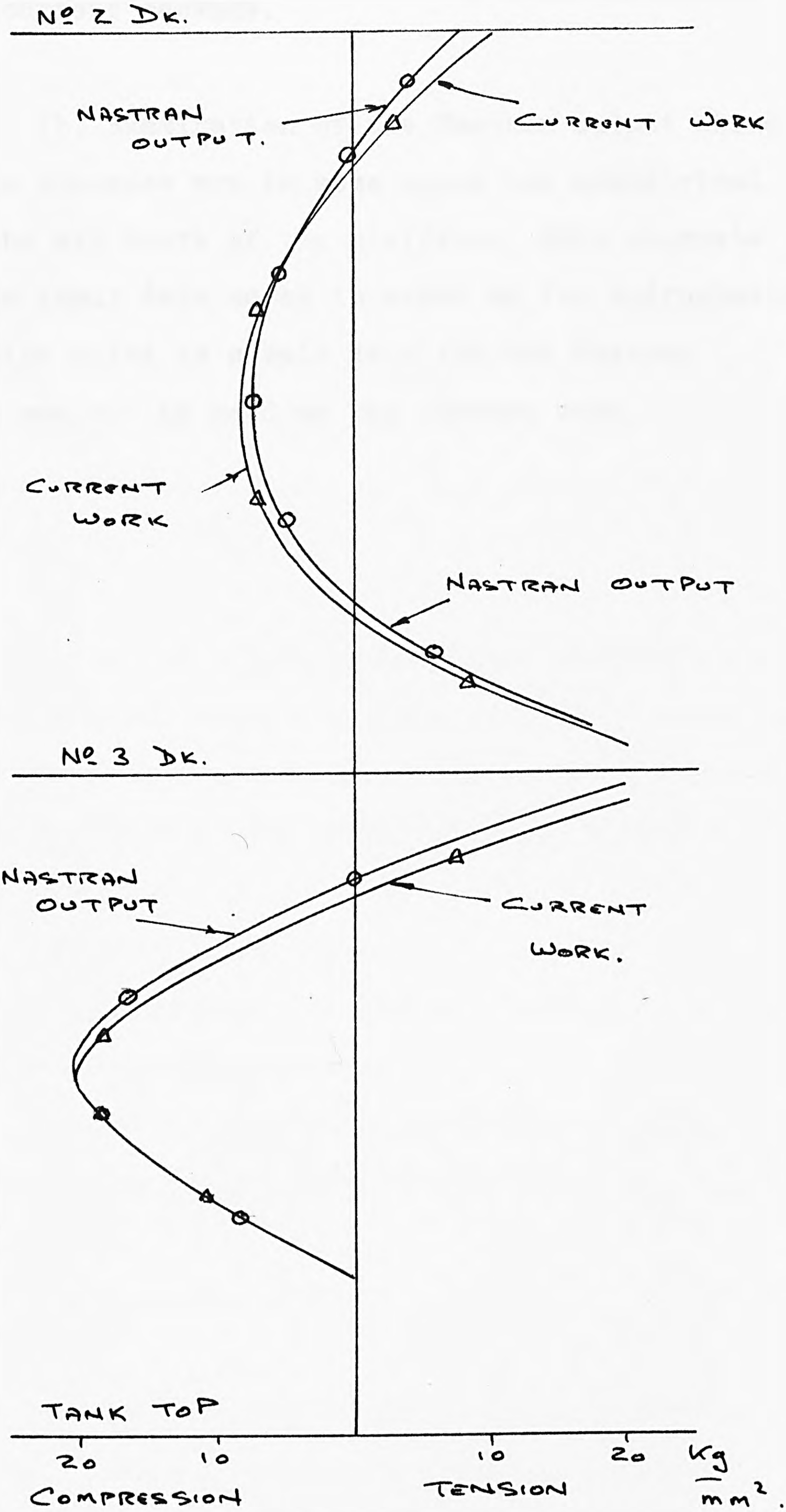
Stresses in stiffener aa 1200 mm off cenre line.

Fig. Test 3.3



Stresses in stiffener bb 5200 mm off centre line.

Fig. Test 3.4



Stresses in stiffener cc 6700 mm off centre line

Fig. Test 3.5

(a) There being no plate bending element in the current package.

(b) Examination of the Nastran output shows that the stresses are in some cases too symmetrical about the mid depth of the stiffener. This suggests that the input data maybe in error or the hydrostatic subroutine added to supply data for the Nastran package was not as good as the current work.

C.6. Test 4

C.6.1 Objectives.

C.6.1.1 As has already been explained the Lusas and Sap 4 packages have different amounts of core space made available for their running. To compare the output for Sap 4 and the current package, it was decided to analyse the largest structure that could be processed by Sap 4 with its smallest array choice selected.

C.6.1.2 The current work contains the facility to deal with thermal loads and loads induced by acceleration of the body as well as the point loads and hydrostatic heads already shown. This analysis deals with three types of loadings:

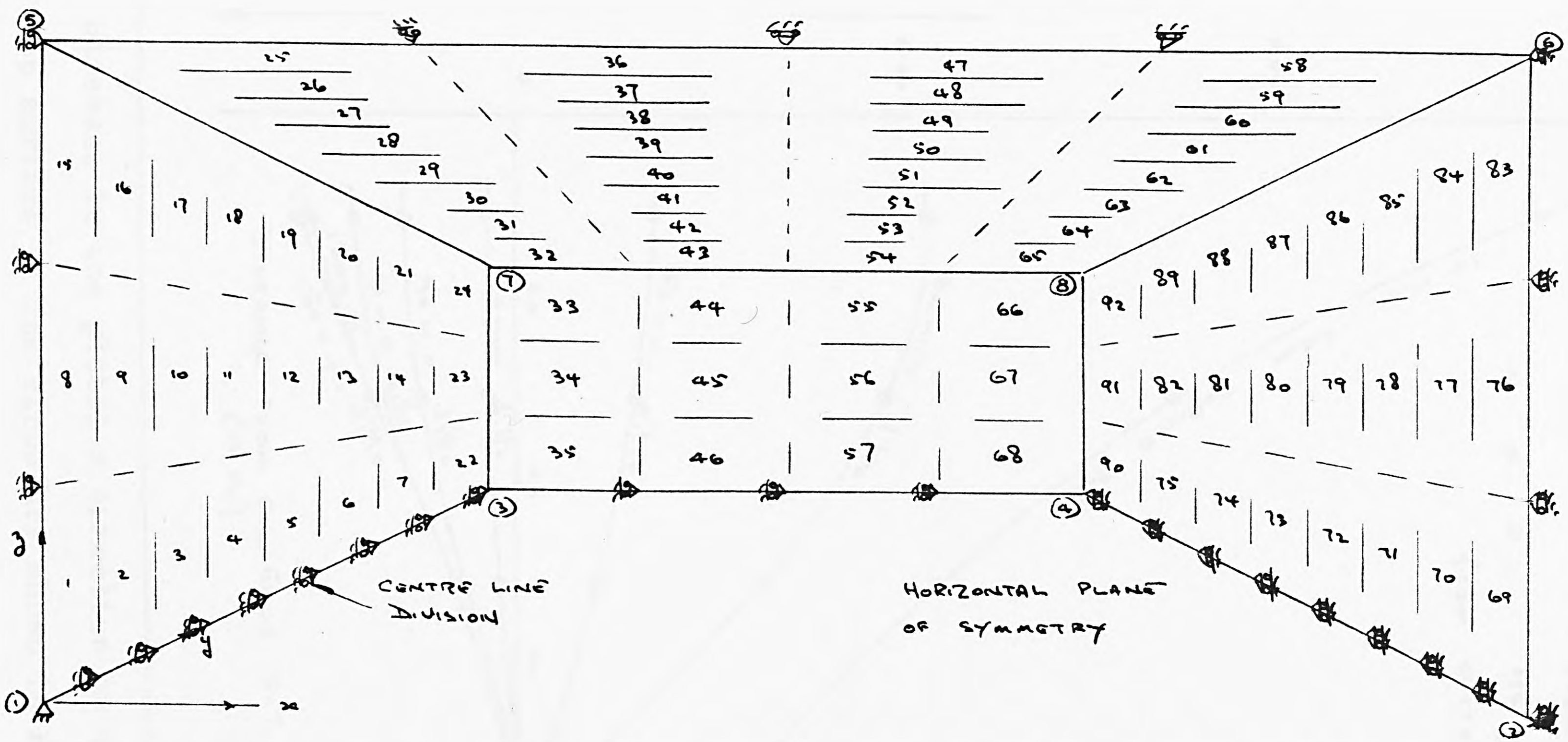
(a) Point load alone.

(b) Point load and thermal loads induced in a heated restrained structure.

(c) Point load and loads due to acceleration of the structure in the three global axes.

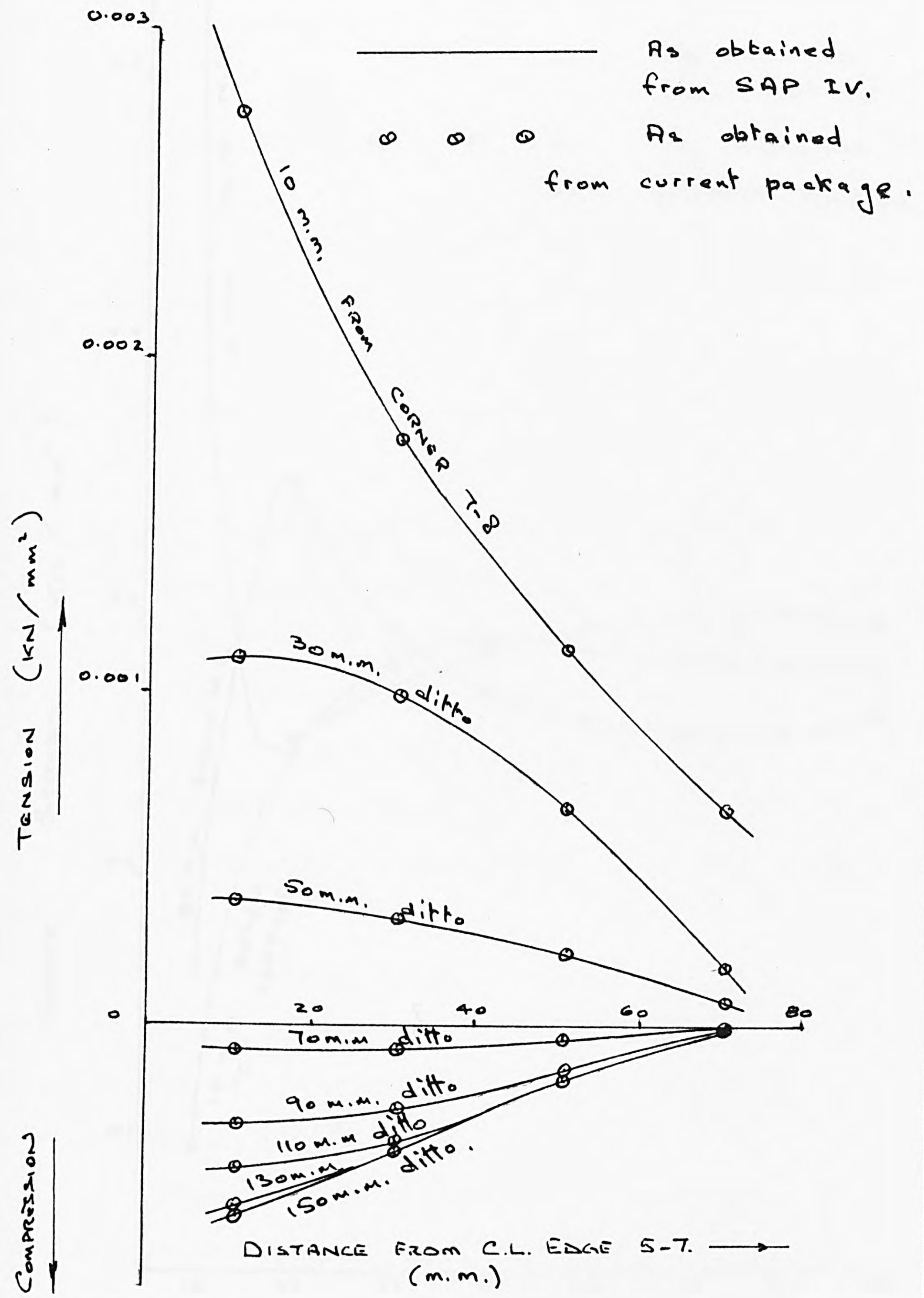
C.6.2 Description.

C.6.2.1 The hypothetical structure was a sheet metal box 320 mm x 160mm x 120mm with a centre division of 320mm x 120mm. There were no beam stiffeners. All the material was 1.5mm thick. The representation consisted of a symmetrical eighth of the structure. All test cases contained a vertical load of -.125 KN applied at the



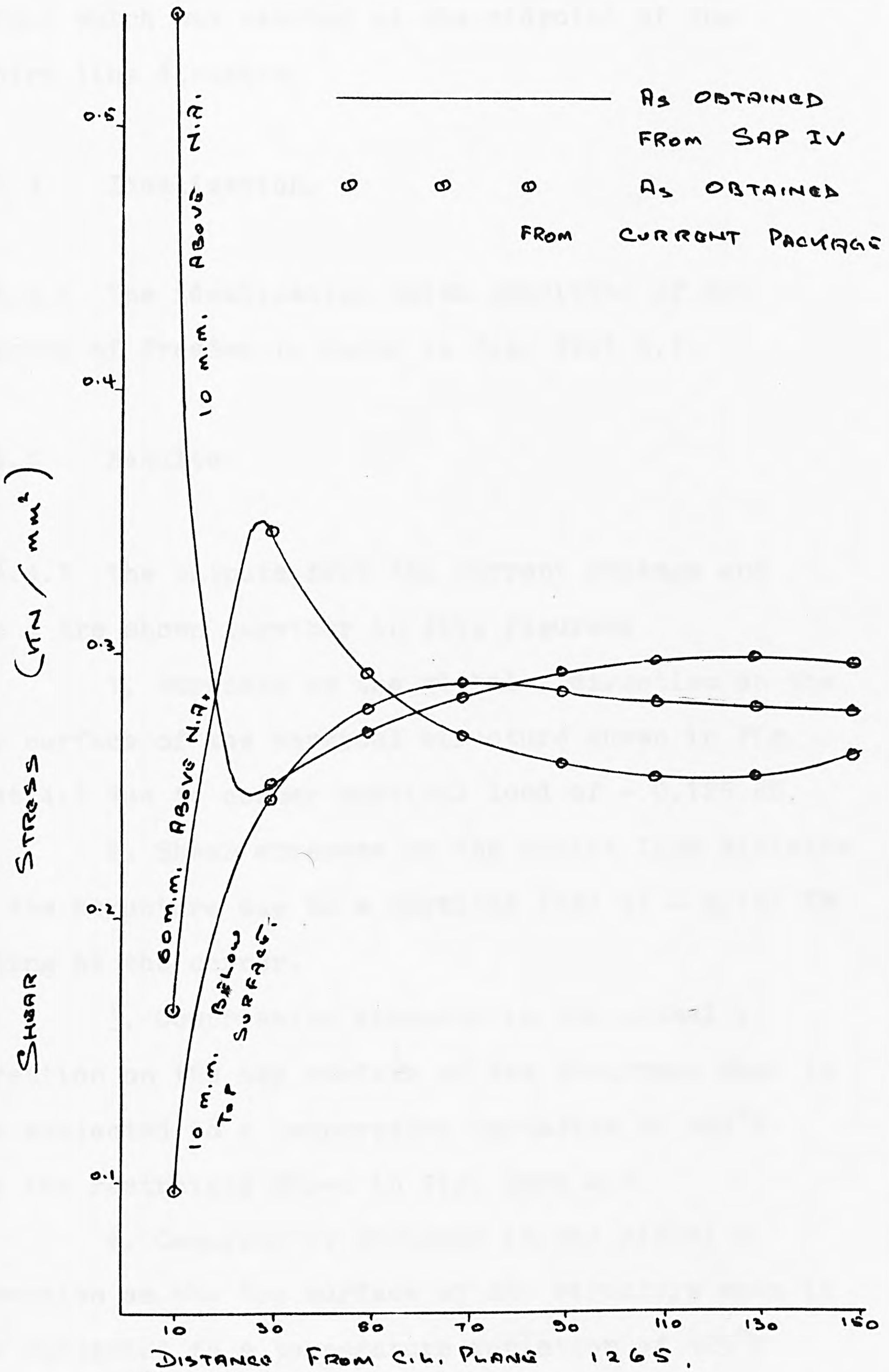
Idealization of 1/8th sheet metal box with centre line division.

Fig. Test 4.1



Stresses in the global x direction on the top surface of the structure shown in Fig. Test 4.1 due to a corner vertical load of -0.125 KN.

Fig. Test 4.2



Shear stresses on centre line division of structure shown in Fig. Test 4.1 due to a vertical load of -0.125 KN acting at the corner.

Fig. Test 4.3

corner which was reacted at the midpoint of the centre line division.

C.6.3 Idealization.

C.6.3.1 The idealization which consisted of 648 degrees of freedom is shown in Fig. Test 4.1.

C.6.4 Results

C.6.4.1 The outputs from the current package and Sap 4 are shown together in five figures;

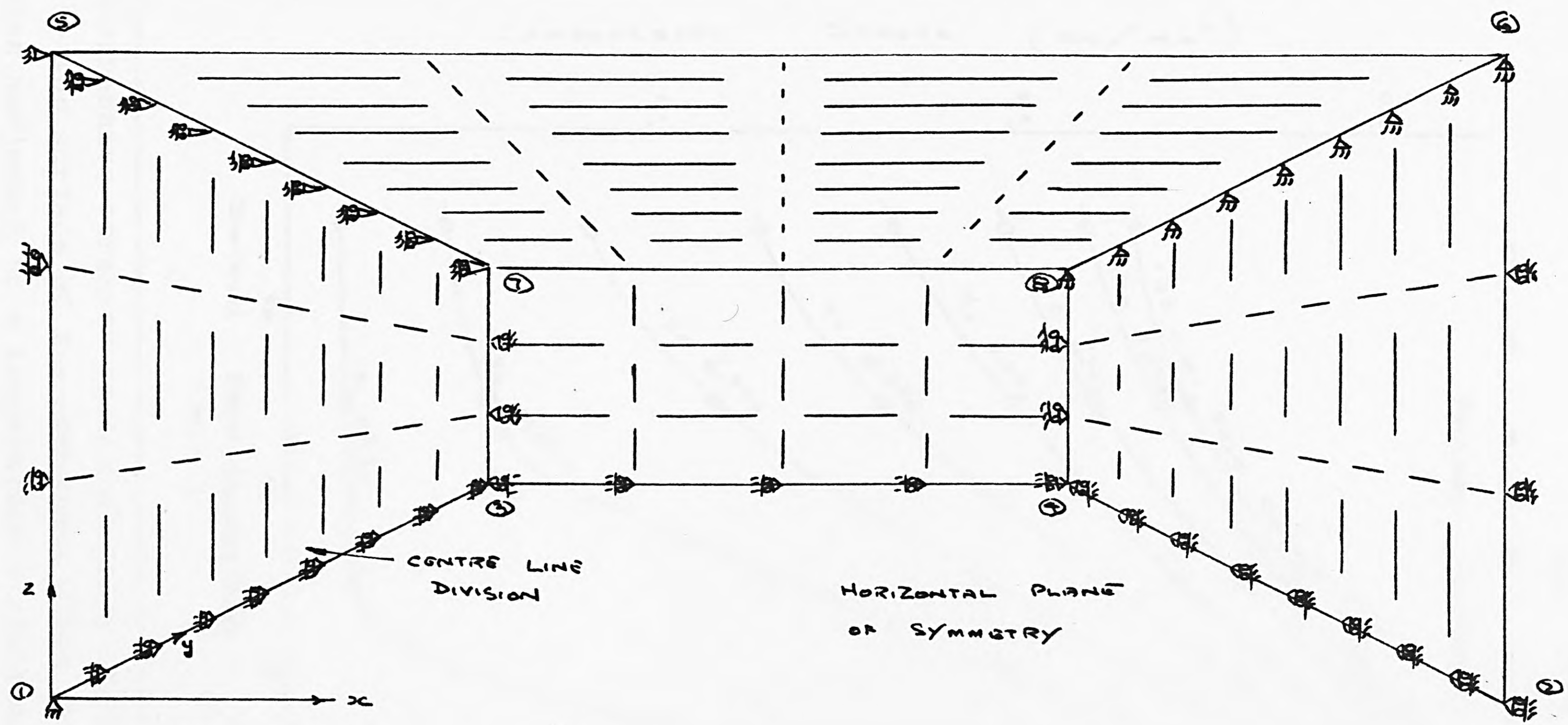
1. Stresses in the global x direction on the top surface of the vertical structure shown in Fig. Test 4.1 due to corner vertical load of - 0.125 KN.

2. Shear stresses on the centre line division of the structure due to a vertical load of - 0.125 KN acting at the corner.

3. Compressive stresses in the global y direction on the top surface of the structure when it was subjected to a temperature variation of +25^{o}C and the restraints shown in Fig. Test 4.4.

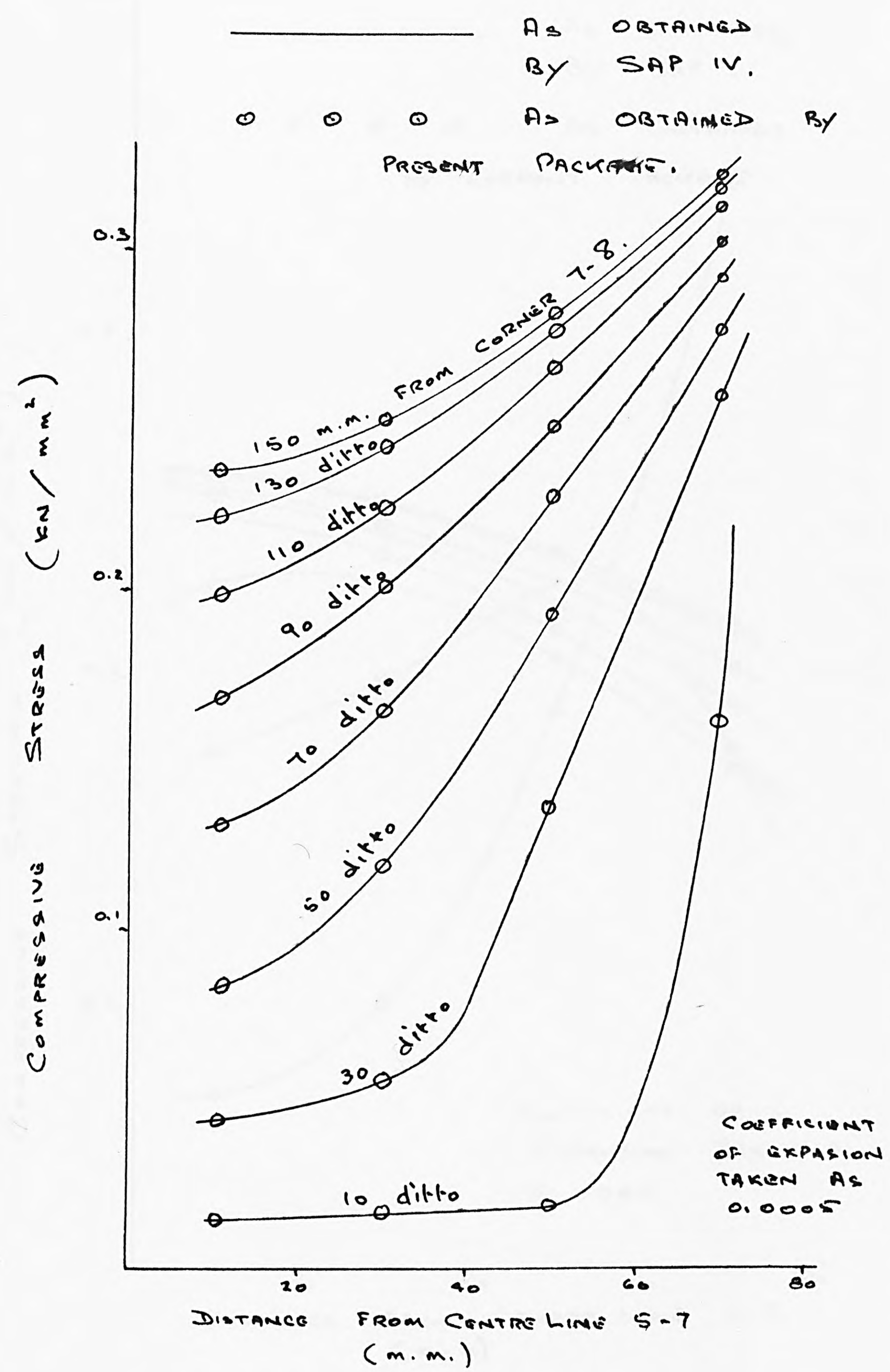
4. Compressive stresses in the global x direction on the top surface of the structure when it was subjected to a temperature variation of +25^{o}C and the restraints shown in Fig. Test 4.4.

5. Tensile stresses due to a point load of - 0.125 KN acting at the stucture's corner and acceleration forces acting on the whole structure.



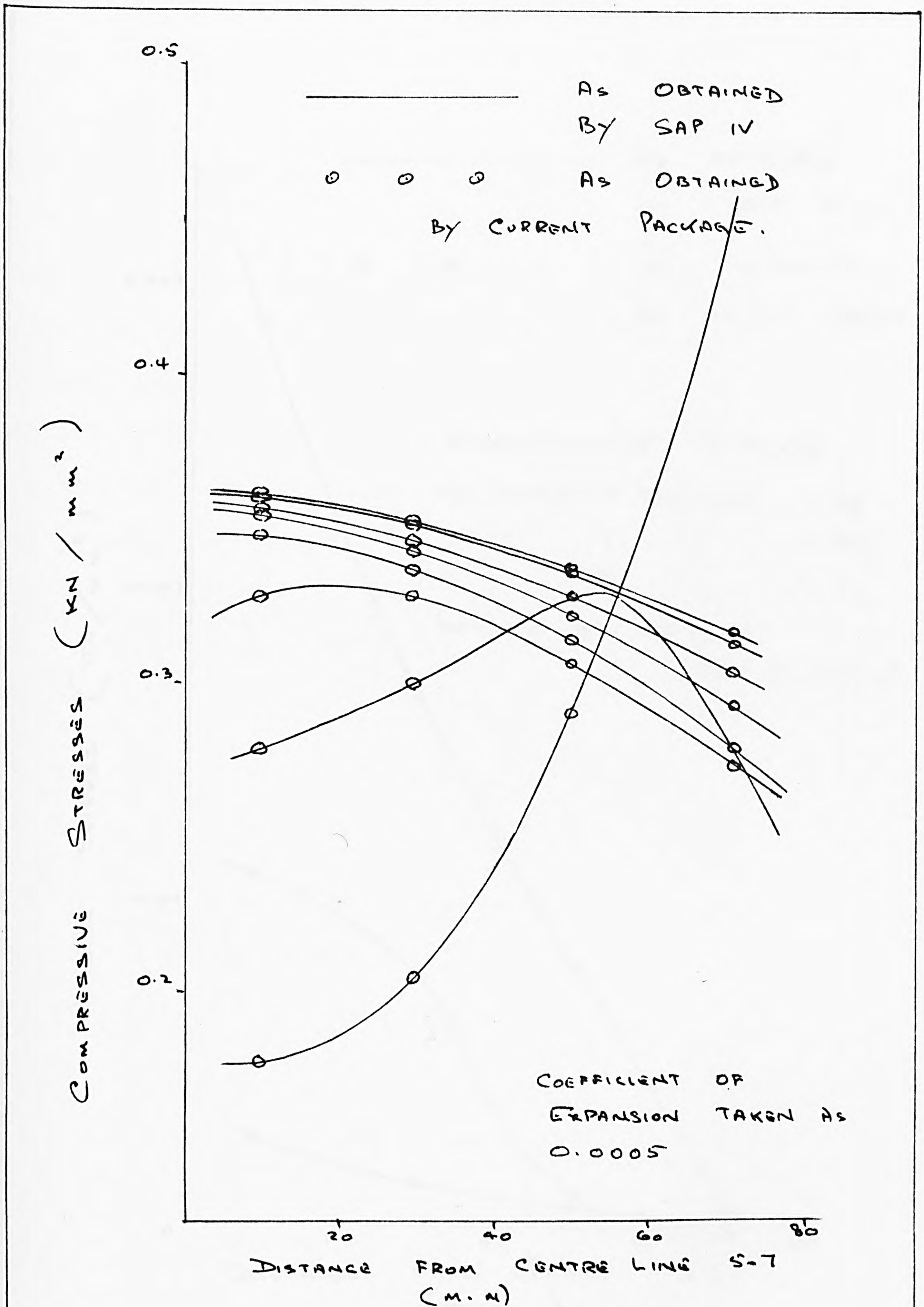
Position and direction of restraints applied to the structure of Fig. Test 4.1 to obtain the thermal stresses given in Fig. Test 4.5

Fig. Test 4.4



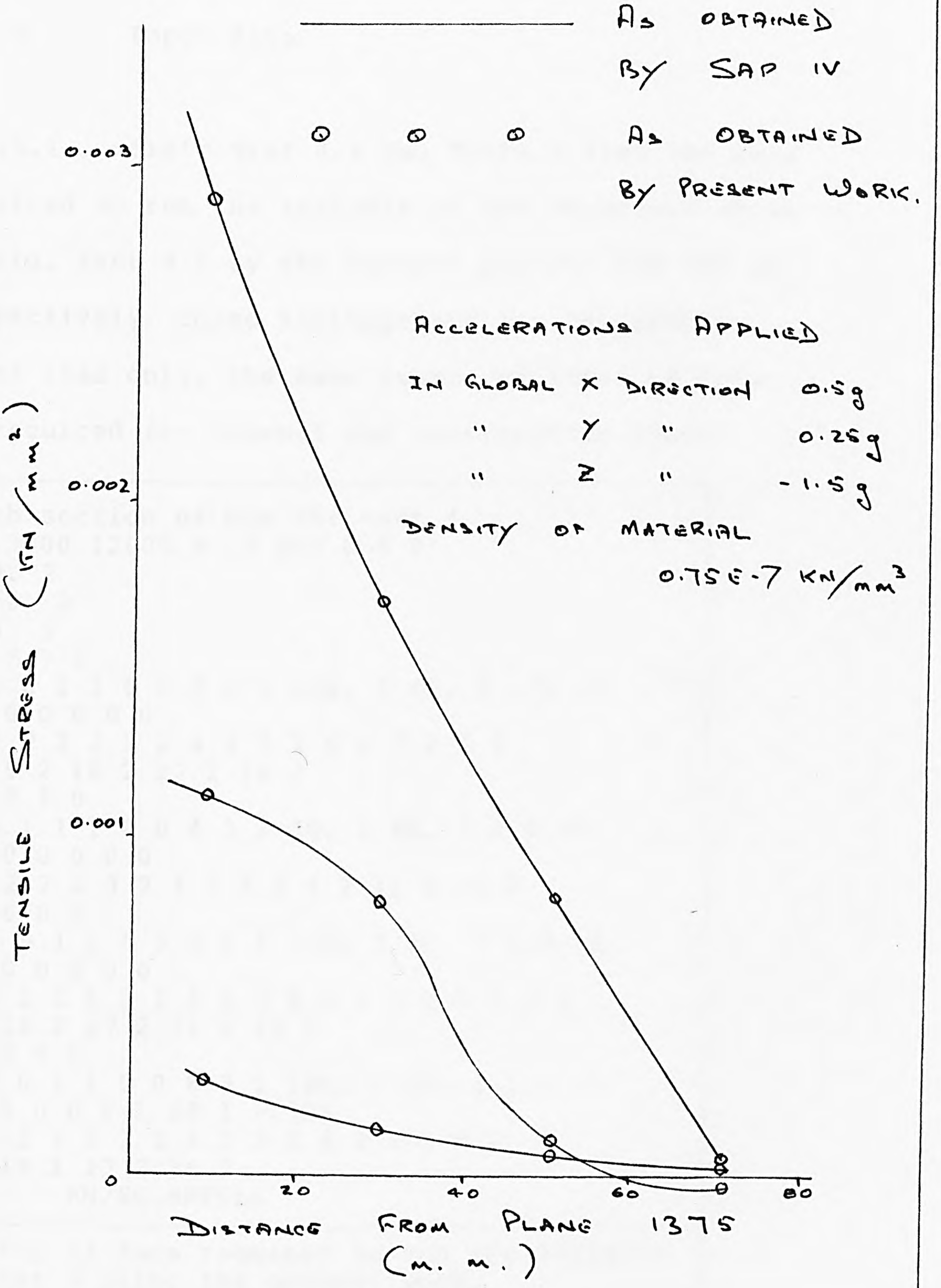
Compressive stresses in the global y direction on the top surface of the structure shown in Fig. Test 4.1 when subjected to a temperature variation of $+25^{0}C$ and the restraints shown in Fig. Test 4.4.

Fig. Test 4.5



Compressive stresses in global x direction on top surface of the structure shown in Fig. Test 4.1 when subjected to a temperature variation of +25°C and the restraints shown in Fig. Test 4.4.

Fig. Test 4.6



Tensile stresses due to a point load of -0.125 KN acting at the strucure's corner and acceleration forces acting on the whole structure of Fig. 4.1

Fig. Test 4.7

C.6.5 Input data

C.6.5.1 Fig's Test 4.8 and Test4.9 list the data required to run the analysis of the structure shown in Fig. Test 4.1 by the current package and SAP IV respectively. These listings are for the single point load only. The same number of lines of data is required for thermal and acceleration loads.

```
1/8th section of box for test 4.
4 0 3000 12000 8 .3 207 0 0 0
1 80. 2
1 160. 2
1 60. 2
3 1 5 7 0
49 0 0 1 1 0 0 8 3 1 160. 1 60. 1 .75 -1
0 0 0 0 0 0 0
13 1 2 2 2 3 2 4 2 5 2 6 2 7 2 8 2
9 1 9 2 18 2 27 2 36 2
3 4 8 7 0
39 0 1 1 1 0 0 4 3 1 80. 1 60. 1 1.5 -1
0 0 0 0 0 0 0
8 1 2 2 2 3 2 4 2 5 2 6 2 11 2 16 2
7 5 6 8 0
51 1 0 1 1 0 0 8 4 1 160. 1 80. 1 1.5 -1
0 0 0 0 0 0 0
14 1 1 2 1 3 1 4 1 5 1 6 1 7 1 8 1 9 1
9 2 18 2 27 2 36 2 45 2
4 2 6 8 0
50 0 0 1 1 0 0 8 3 1 160. 1 60. 1 1.5 -1
0 0 0 0 0 0 1 28 1 -.125
12 1 2 2 2 3 2 4 2 5 2 6 2 7 2 8 2
9 2 18 2 27 2 36 2
KN       KN/SQ.MMFULL
```

Listing of data required to run the analysis in Test 4 using the present work.

Fig. Test4.8

108 NODES FORMING 1/8 SECTION OF BOX WITH C.L. WEB.

108	1	1								
1	1	1	1	1	1	1	0	0		
2	1	1		1	1	1	0	20	0	1
8	1	1		1	1	1		140		1
9	1	1		1	1	1			20	1
10	1			1	1	1		20	20	1
16	1			1	1	1		140	20	1
17	1	1		1	1	1			40	1
18	1			1	1	1		20	40	1
24	1			1	1	1		140	40	1
25	1	1		1	1	1			60	1
26	1			1	1	1		20	60	1
33	1			1	1	1		160	60	1
36	1	1		1	1	1		160		1
37		1	1	1	1	1	20		60	1
38			1	1	1	1	20	20	60	1
44			1	1	1	1	20	140	60	1
45			0	1	1	1	20	160	60	1
46		1		1	1	1	20	160	40	1
48	1	1		1	1	1	20	160	0	1
49		1	1	1	1	1	40		60	1
50			1	1	1	1	40	20	60	1
56			1	1	1	1	40	140	60	1
57			0	1	1	1	40	160	60	1
58		1		1	1	1	40	160	40	1

Part listing of data required to run analysis of Test 4 using SAP IV.

Fig. Test 4.9 (a)

60	1	1		1	1	1	40	160	0	1	
61		1	1	1	1	1	60		60	1	
62			1	1	1	1	60	20	60	1	
68			1	1	1	1	60	140	60	1	
69			0	1	1	1	60	160	60	1	
70		1		1	1	1	60	160	40	1	
72	1	1		1	1	1	60	160	0	1	
73	1	1		1	1	1	80	0		1	
74	1	1		1	1	1	80	20		1	
80	1	1		1	1	1	80	140		1	
81	1	1		1	1	1	80	0	20	1	
82	1			1	1	1	80	20	20	1	
88	1			1	1	1	80	140	20	1	
89	1	1		1	1	1	80	0	40	1	
90	1			1	1	1	80	20	40	1	
96	1			1	1	1	80	140	40	1	
97	0	1		1	1	1	80	0	60	1	
98	0			1	1	1	80	20	60	1	
104	0			1	1	1	80	140	60	1	
105				1	1	1	80	160	60	1	
106				1	1	1	80	160	40	1	
108	1	1		1	1	1	80	160	0	1	
3	92	1									
1											
			207		207		207	0.3	0.3	0.3	79.615
		0									
		0									

Continuation of data for Test4 using SAP IV.

Fig. Test 4.9(b)

		0							
		0							
		0							
1	2	1	9	10	1		20	1	0.75
8	10	9	17	18	1		20	1	0.75
15	18	17	25	26	1		20	1	0.75
22	36	8	16	35	1		20	1	0.75
23	35	16	24	34	1		20	1	0.75
24	34	24	32	33	1		20	1	0.75
25	26	25	37	38	1		20	1	1.5
36	38	37	49	50	1		20	1	1.5
47	50	49	61	62	1		20	1	1.5
58	62	61	97	98	1		20	1	1.5
69	74	73	81	82	1		20	1	1.5
76	82	81	89	90	1		20	1	1.5
83	90	89	97	98	1		20	1	1.5
90	108	80	88	107	1		20	1	1.5
91	107	88	96	106	1		20	1	1.5
92	106	96	104	105	1		20	1	1.5
105	1					-0.125			
	1		1		1	1			
	0								
	0								
	0								
	0								
	0								

Final part of the data required to run the analysis of Test 4 using the SAP IV package.

Fig. Test 4.9(c)

C.6.6 Conclusions.

C.6.6.1 It can be seen from the set of figures giving the results from both the current package and the very well tested Sap 4 package that there is total harmony of results to such an extent that all the result points are coincident.

C.6.6.2 The core space used by SAP IV for this problem was 95K, whilst the current package used 64K.

C.7 Test 5

C.7.1 Objectives.

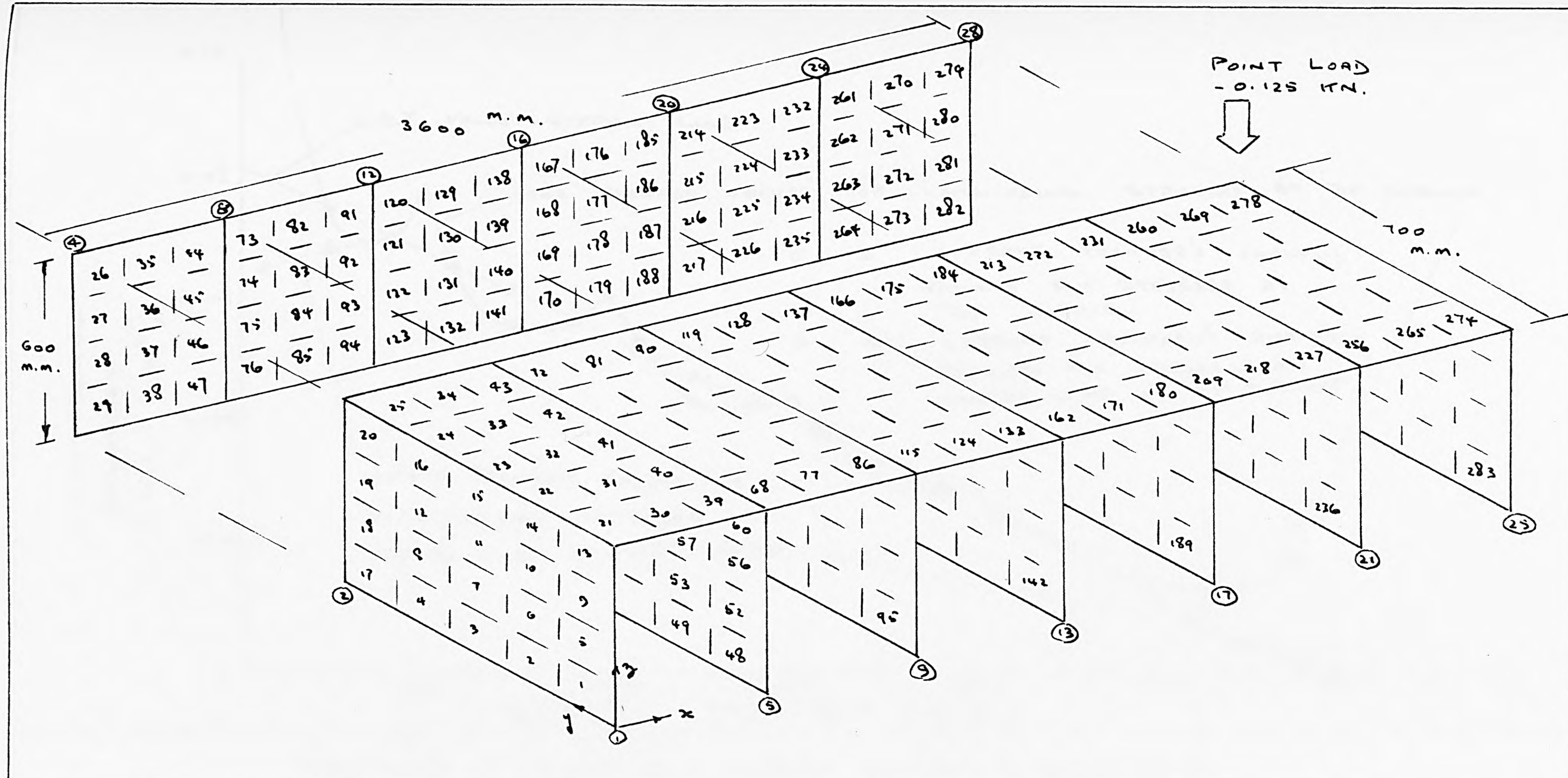
C.7.1.1 To compare the largest analysis which could be undertaken using the Lusas package and the maximum amount of core space normally allowed by the City University for its running and the current work using only a fraction of that core space.

C.7.2 Description.

C.7.2.1 The structure was an aluminium alloy box beam, having a total length of 7200mm and a cross section of 1200mm x 1400mm. The beam was transversely subdivided into 12 boxes of equal length. A load of 1KN was assummed to be acting vertically downwards through the beam's centre of gravity and it was considered simply supported at each end. All the material was taken to be of a constant unit thickness.

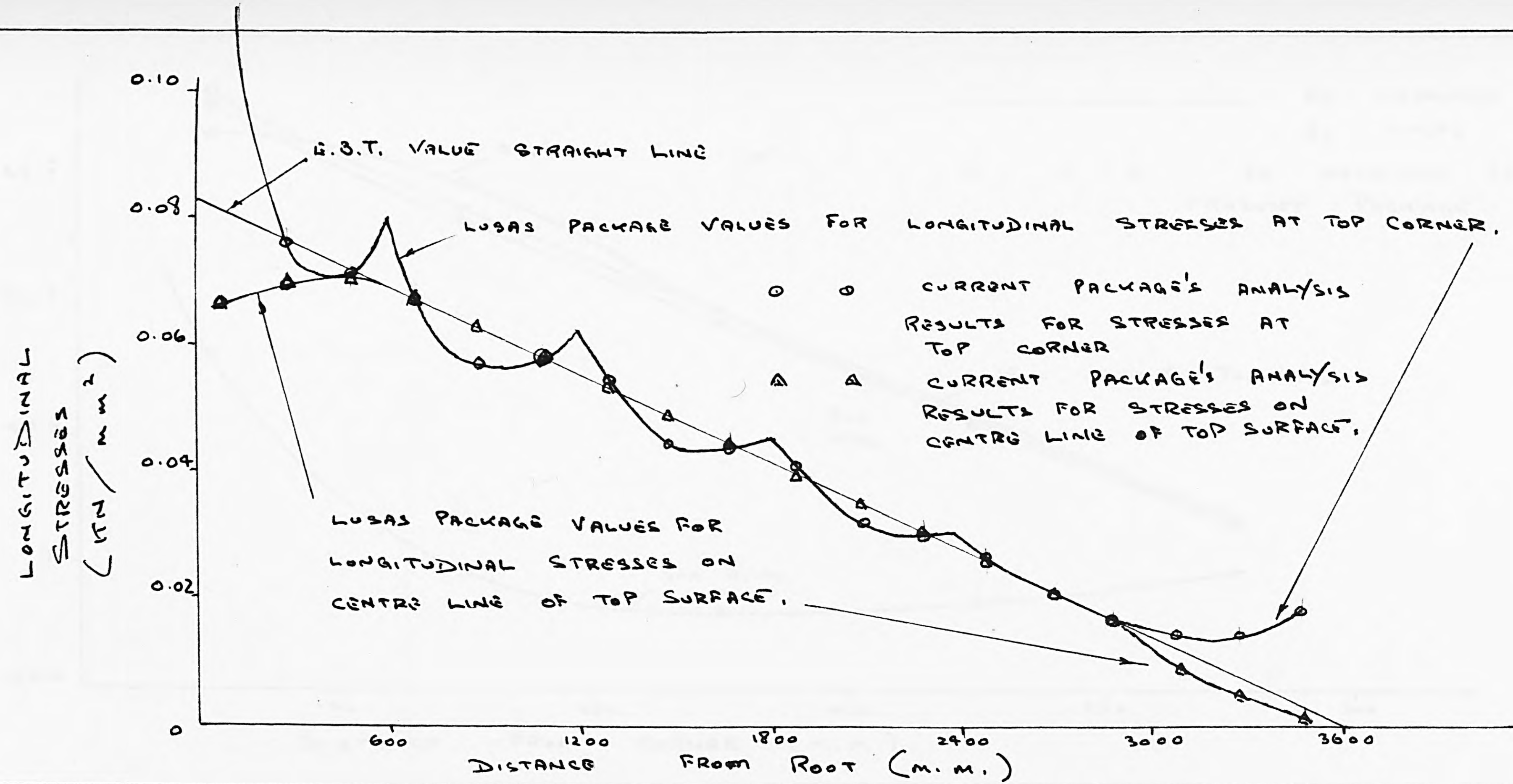
C.7.3 Idealization.

C.7.3.1 The model used for the analysis represented one eighth of the symmetrical structure and is shown in Fig. Test 5.1. This figure shows the numbering of the corners of the panels used to define the beam for the current work and also the numbering of the plate elements as used by the Lusas package. 302 plate elements were used, each having 4 nodes per plate and 2 degrees of freedom per node.



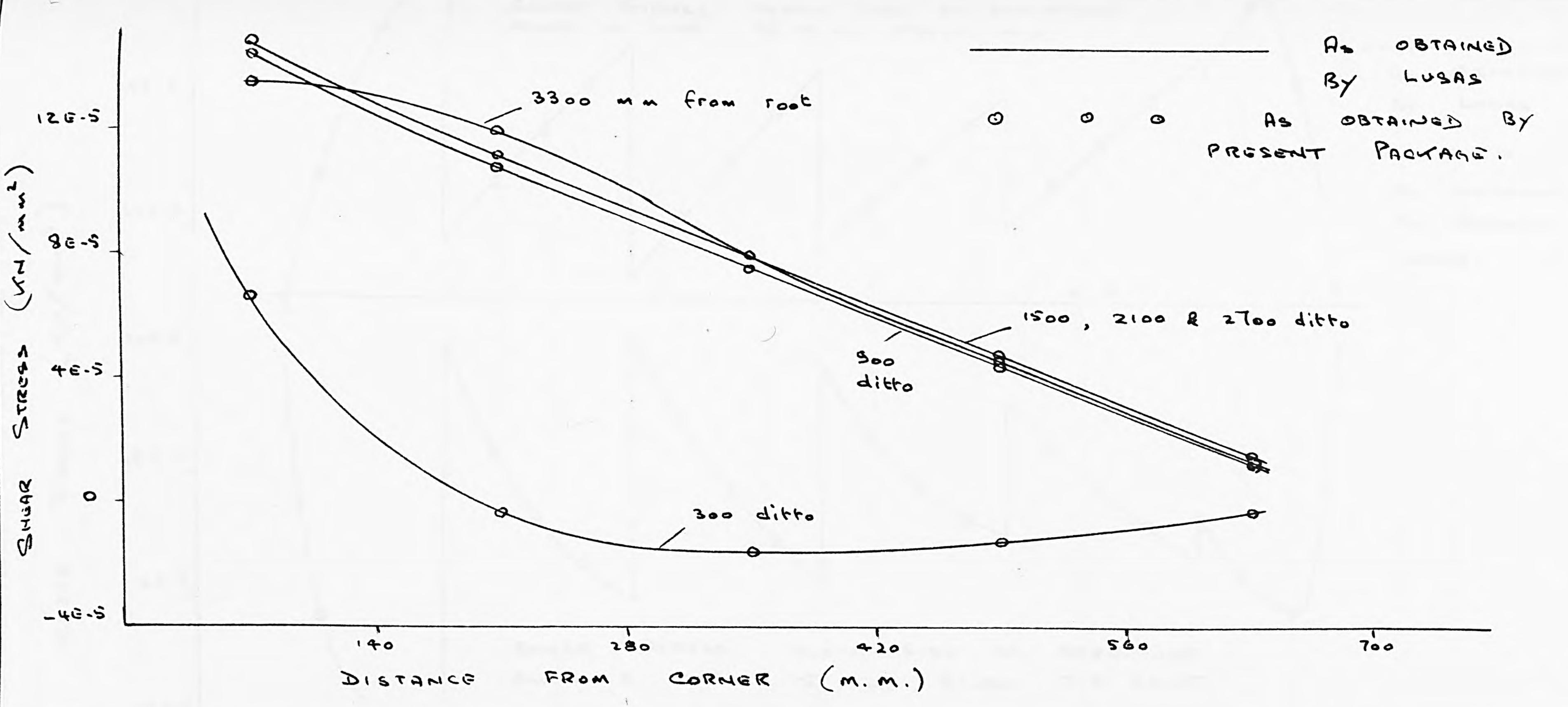
Hypothetical beam structure used for comparing the current work and Lusas.

Fig. Test 5.1



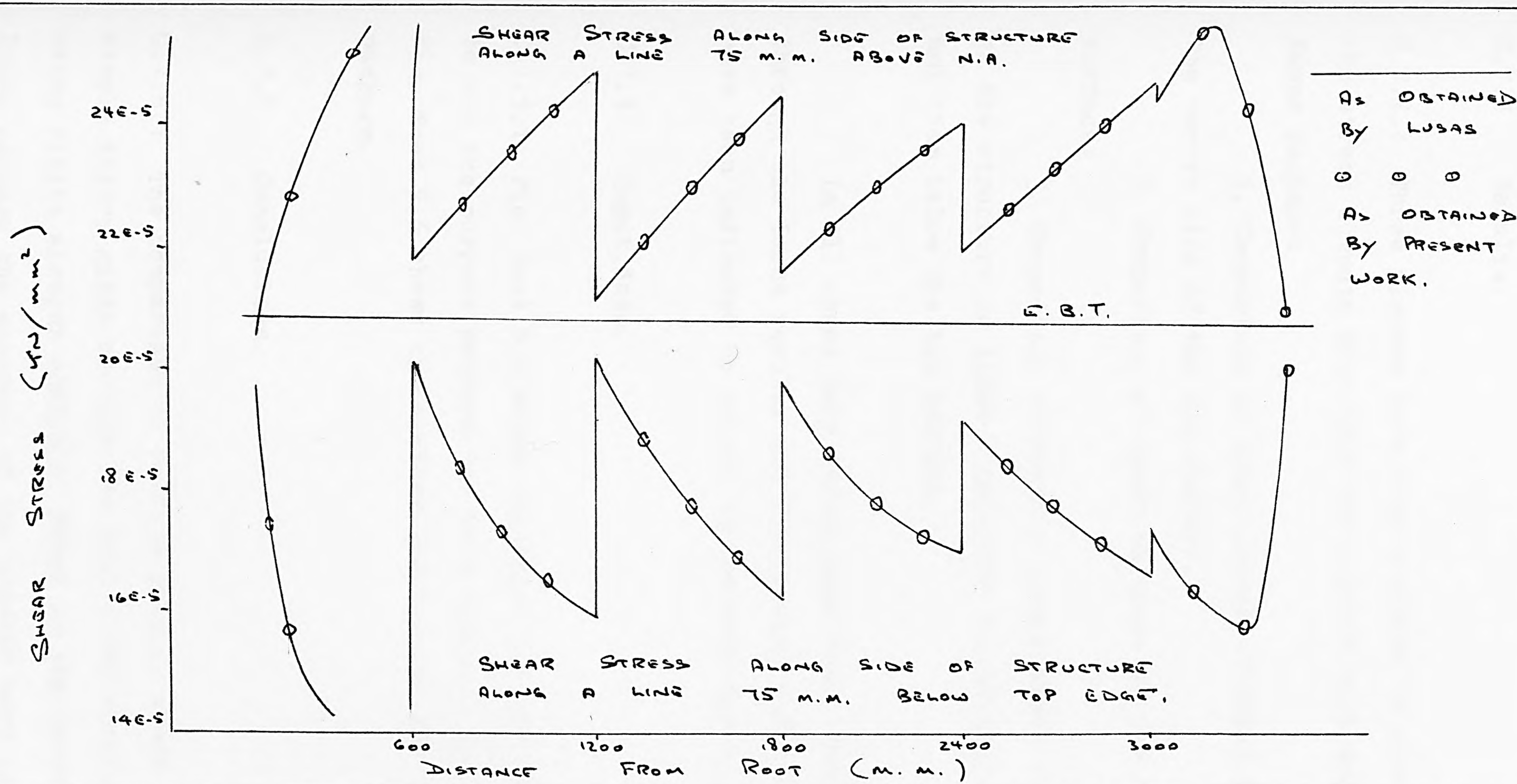
Comparison of longitudinal stresses obtained by using Lusas current work and E.B.T. for the structure shown in Fig. Test 5.1

Fig. Test 5.2



Comparison of the shear stresses obtained by using Lusas and the current work on the top surface of the structure shown in Fig. Test5.1

Fig. Test 5.3



Comparison of the shear stresses obtained by using Lusas and the current package on the side of the structure shown in Fig. Test 5.1

Fig. Test 5.4

C.7.4 Results.

C.7.4.1 Three figures have been prepared to show the stress outputs from both the current work and the Lusas package:

1. Comparison of longitudinal stresses on the centre line of the top surface.

2. Comparison of shear stresses on the top surface.

3. Comparison of shear stresses along the side of the structure on lines 75mm above the neutral axis and 75mm below the top surface.

In all cases fair curves have been drawn through the Lusas results and the current work results have been indicated by points on the same figure.

C.7.5 Input data.

C.7.5.1 Fig. Test 5.5 shows the input for the data to run the current package for this analysis and Fig. Test 5.6 shows the similar listing for the Lusas package.

C.7.6 Conclusions.

C.7.6.1 The comparison of results clearly shows that simple theory gives results far from those obtained using finite element analysis. Based on the proven Lusas package the accuracy of the present work is

```
Box in 6 bays for Test 5
19 0 3000 12000 28 .3333 72.4 0 0 0
1 700. 2
1 600. 2
1 3600. 7
2 1 3 4 0
45 0 0 1 1 0 0 5 4 1 700. 1 600. 1 1 -1 0 0 0 0 0 0 0
11 1 2 2 2 3 2 4 2 5 2 6 1 6 2 12 2 18 2 24 2 30 2
2 6 8 4 0
39 0 1 1 1 0 0 3 4 1 600. 1 600. 1 1 -1 0 0 0 0 0 0 0
8 1 2 2 2 3 2 4 2 5 2 9 2 13 2 17 2
3 7 8 4 0
43 0 1 1 1 0 0 3 5 1 600. 1 700. 1 1 -1 0 0 0 0 0 0 0
10 1 1 1 2 2 1 3 1 4 1 5 2 9 2 13 2 17 2 21 2
6 5 7 8 1
43 0 0 1 1 0 0 5 4 1 700. 1 600. 1 1 -1 0 0 0 0 0 0 0
10 1 2 2 2 3 2 4 2 5 2 6 2 12 2 18 2 24 2 30 2
6 10 12 8 2
31 0 1 1 1 0 0 3 4 1 600. 1 600. 1 1 -1 0 0 0 0 0 0 0
4 1 2 2 2 3 2 4 2
7 11 12 8 3
31 0 1 1 1 0 0 3 5 1 600. 1 700. 1 1 -1 0 0 0 0 0 0 0
4 1 1 2 1 3 1 4 1
10 9 11 12 -1
10 14 16 12 -2
11 15 16 12 -3
14 13 15 16 -1
14 18 20 16 -2
15 19 20 16 -3
18 17 19 20 -1
18 22 24 20 -2
19 23 24 20 -3
22 21 23 24 -1
22 26 28 24 -2
23 27 28 24 -3
26 25 27 28 0
46 0 0 1 1 0 0 5 4 1 700. 1 600. 1 1 -1 0 0 0 0 0 0
1 25 1 -.125 10 1 2 2 2 3 2 4 2 5 2 6 2 12 2 18 2 24
2 30 2
MM      KN/SQ.MMFULL
```

Listing of data required to run the analysis of Test 5 using the present work.

Fig. Test 5.5

```
PROBLEM TITLE 6 BAY BCX FOR TEST 5
UNITS KN MM
OPTION 14
OPTION 15
OPTION 34
SM14 ELEMENT TOPOLOGY
FIRST 1 2 1 6 7
INC 1 1 1 1 1 4
INC 4 5 5 5 5 4
INC 47 50 50 50 50 7
FIRST 17 30 5 10 29
INC 1 -1 5 5 -1 4
INC 47 50 50 50 50 7
FIRST 21 21 31 32 22
INC 1 1 1 1 1 9
INC 9 10 10 10 10 2
INC 47 50 50 50 50 6
FIRST 39 41 71 72 42
INC 1 1 1 1 1 9
INC 47 50 50 50 50 6
NODE COORDINATES
330 3600 700 0
FIRST 1 0 0 0
INC 1 0 140 0 5
INC 50 600 0 0 7
FIRST 21 0 0 600
INC 1 0 140 0 6
INC 50 600 0 0 7
FIRST 30 0 700 0
INC 50 600 0 0 6
INC 10 200 0 0 3
FIRST 31 200 0 600
INC 50 600 0 0 6
INC 5 0 700 0 2
FIRST 37 200 700 300
INC 50 600 0 0 6
INC 10 200 0 0 2
FIRST 41 400 0 600
INC 50 600 0 0 6
INC 5 0 700 0 2
SPACING
FIRST 1 21 5 4*1
INC 1 1 (5)
INC 50 50 (7)
FIRST 31 281 50 5*1
```

First part of the listing of data required to run Test 5 using the Lusas package.

Fig. Test 5.6(a)

```
INC 10 10 (2)
INC 6 6 (2)
FIRST 31 36 1 5*1
INC 50 50 (6)
INC 10 10 (2)
FIRST 26 30 1 4*1
INC 50 50 (7)
FIRST 36 40 1 4*1
INC 10 10 (2)
INC 50 50 (6)
SMI4 GEOMETRIC PROPERTIES
1 302 1 1. 1. 1. 1.
MATERIAL PROPERTIES
1 302 1 72.4 .3333
MEMBRANE RIGIDITIES
1 302 1 8.145E1 8.145E1 2.715E1 2.715E1 0
SUPPORT NODES
1 0 0 R R R
2 5 1 R R F
6 21 5 R R F
FIRST 56 66 5 R R F
INC    50 50 (6)
71 321 50 F R F
FIRST 51 55 1 R R F
INC 50 50 (6)
30 330 50 R R F
26 29 1 R F F
FIRST 7 10 1 R F F
INC 5 5 (3)
INC 50 50 (7)
22 25 1 R F F
FIRST 31 41 10 F R R
INC    50 50 (6)
FIRST 40 50 10 R R F
INC    50 50 (6)
FIRST 32 35 1 F F R
INC 10 10 (2)
INC 50 50 (6)
FIRST 37 39 1 F R F
INC 10 10 (2)
INC 50 50 (6)
LOAD CASES
CONCENTRATED LOAD
326 0 0 0 0 -.125
END
```

Final listing part for the data required to run Test 5 using the Lusas package.

Fig. Test 5.6(b)

completely validated.

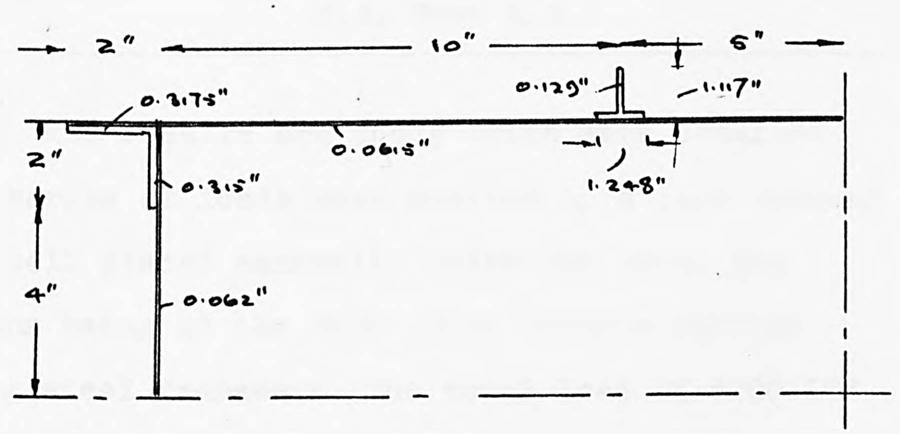
C.7.6.2 The core space used by Lusas for this problem was 120K, wkilst the current package used 65K.

C.8 Test 6

C.8.1 Objectives.

C.8.1.1 The analysis results so far reported in this appendix have all been compared with the results obtained from other structural analysis packages or simple bending theory. It is the objective of this test to compare the results of an analysis using the present work with those obtained from strain gauge readings. At the same time using less core than that which would have been used by other packages having the same capabilities.

C.8.2 Description.

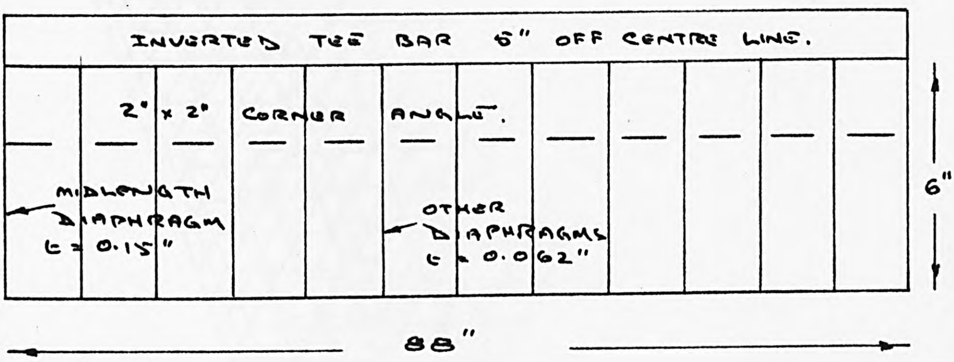


Quarter section of box beam used in test 6.

Fig. Test 6.1

C.8.2.1 All the details of the structure and the test report were supplied by the Aeronautical Department of the City University. All the dimensions and units for the results are in inches and lbf/in^2.

C.8.2.2 The structure consisted of an aluminium alloy box beam of 176" length and having a quarter section as shown in Fig. Test 6.1. The beam was fitted with twenty five transverse web plates or diaphragms. A quarter longitudinal section is shown in Fig. Test 6.2



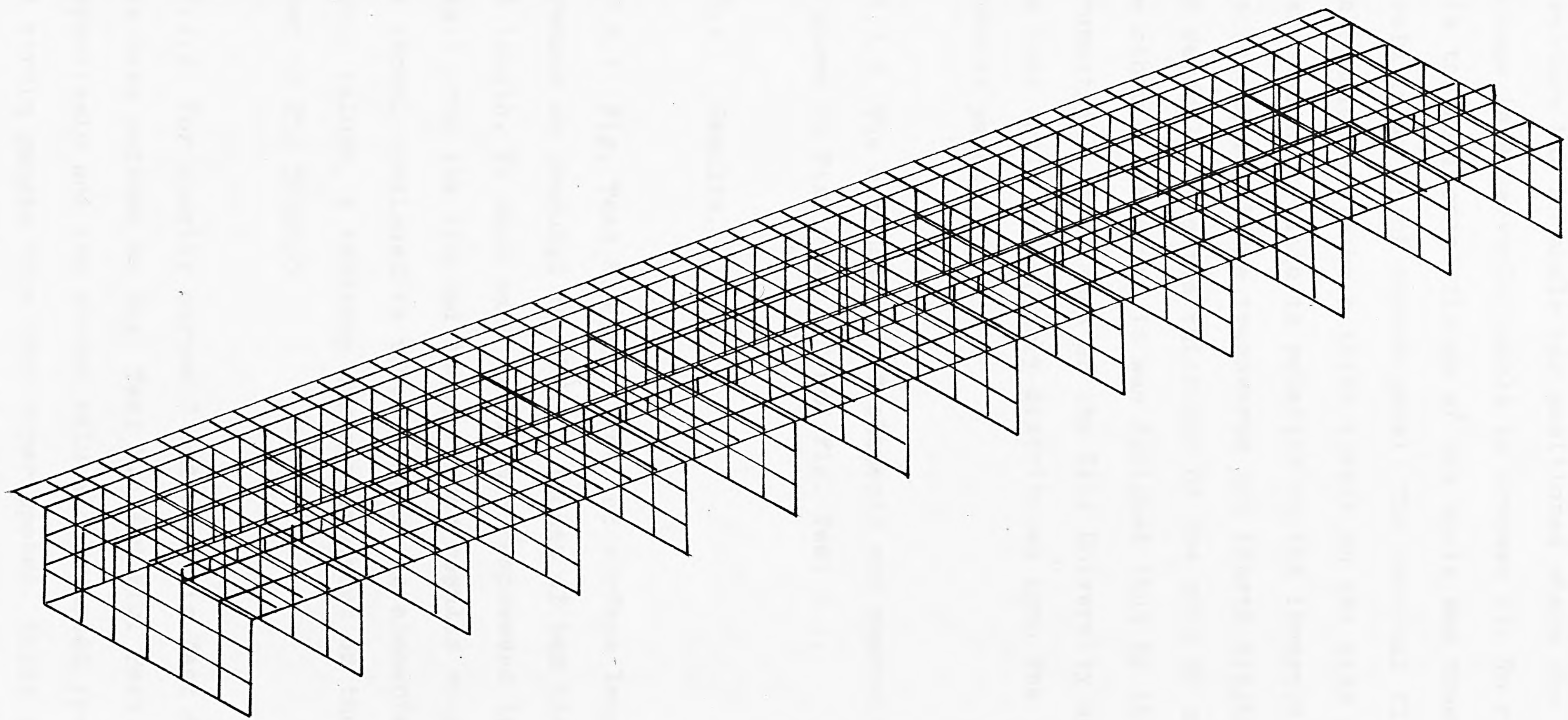
Disposition of equally spaced diaphragms

Fig. Test 6.2

C.8.2.3 The results are those which were obtained when a series of loads were applied by a jack through a load cell placed centrally under the beam. The reactions being at the four upper corners through a strong steel framework. The total load of 4000 lbf was applied in eight equal steps of 500 lbf.

C.8.3 Idealization.

C.8.3.1 This package is unable to accommodate a structure having a stiffener on an edge of a panel, if that edge is not a free edge. The section of this



Idealization of the box beam used in test 6.

Fig. Test 6.3

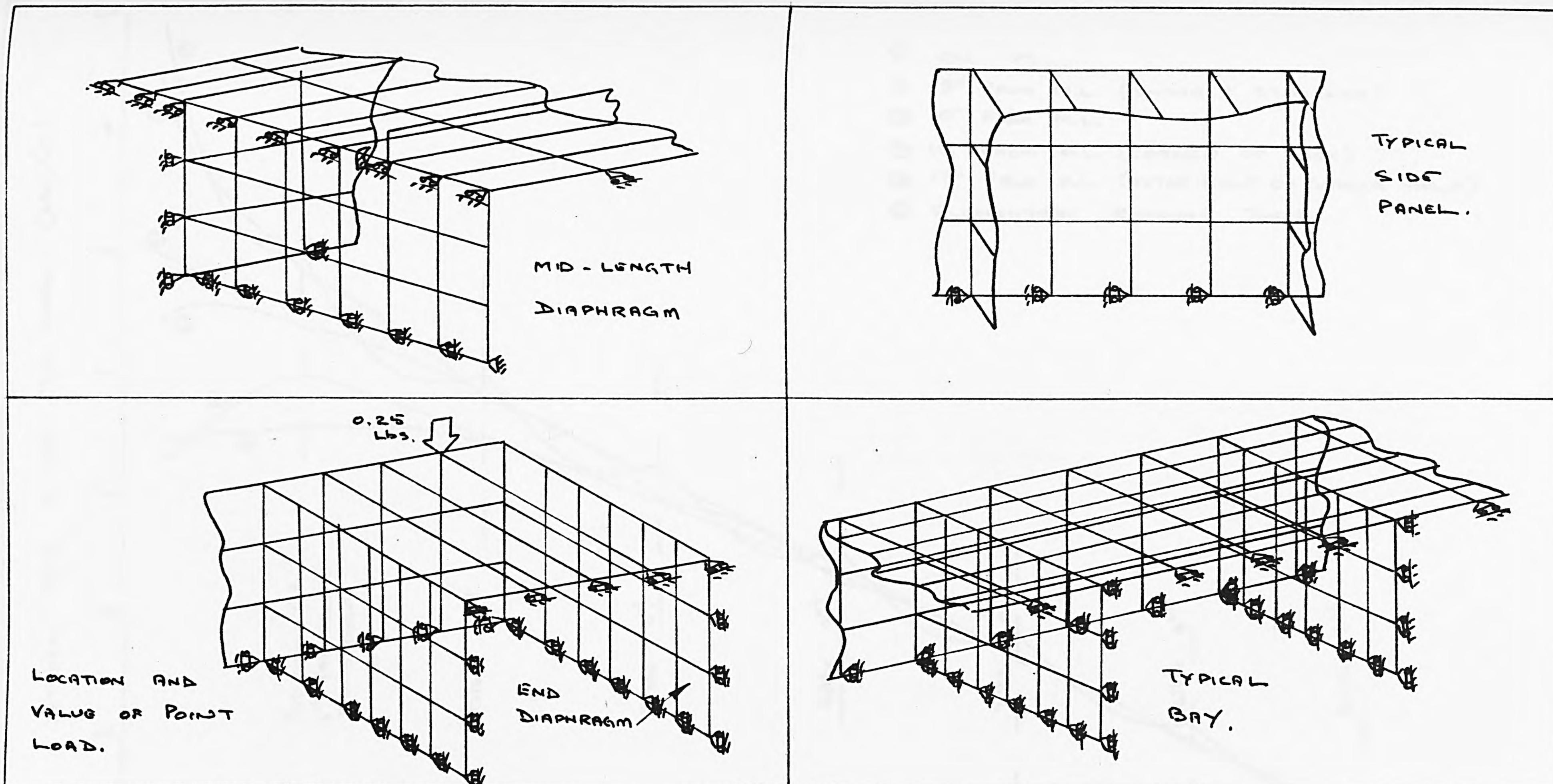
structure has an angle bar positioned where the package is apparently unable to process it. To overcome this the horizontal flange of the angle was treated as a relatively thick narrow panel. The vertical flange was considered to be a thick insert on the side panel. There was no trouble in relation to the inverted tee bar. The thickness of the transverse mid length diaphragm was several times the thickness of the skin or any of the other diaphragms. It was designed thus by the Aeronautical Department of the City University so that the load could be suitably distributed into the adjacent panels.

C.8.3.2 The arrangement of elements and restraints is shown in Fig. Test 6.3 and Fig. Test 6.4.

C.8.4 Results.

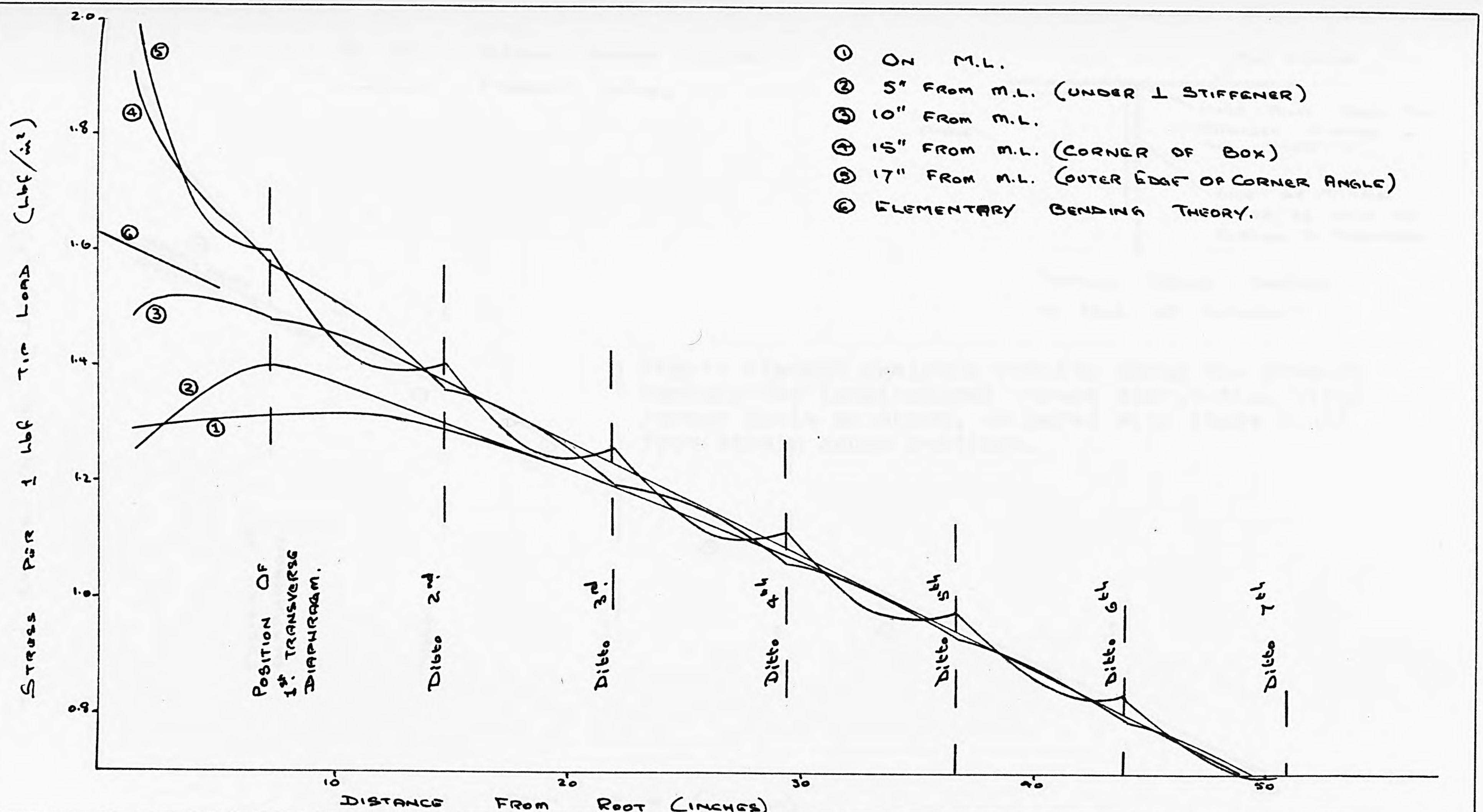
C.8.4.1 Fig. Test 6.5 gives the top surface longitudinal stresses as computed for the 50 inches of box beam from mid length. To show more would have compressed too much detail into the limited space. Those results which are not shown, continued to move towards the elementary theory values, a tendancy already apparent on the part drawn in Fig Test6.5

C.8.4.2 For clarity curves 2 and 4 of Fig Test 6.5 have been redrawn as Fig. Test 6.6 and Fig. Test 6.7 respectively and the stress values calculated from the strain gauges have been superimposed. These show

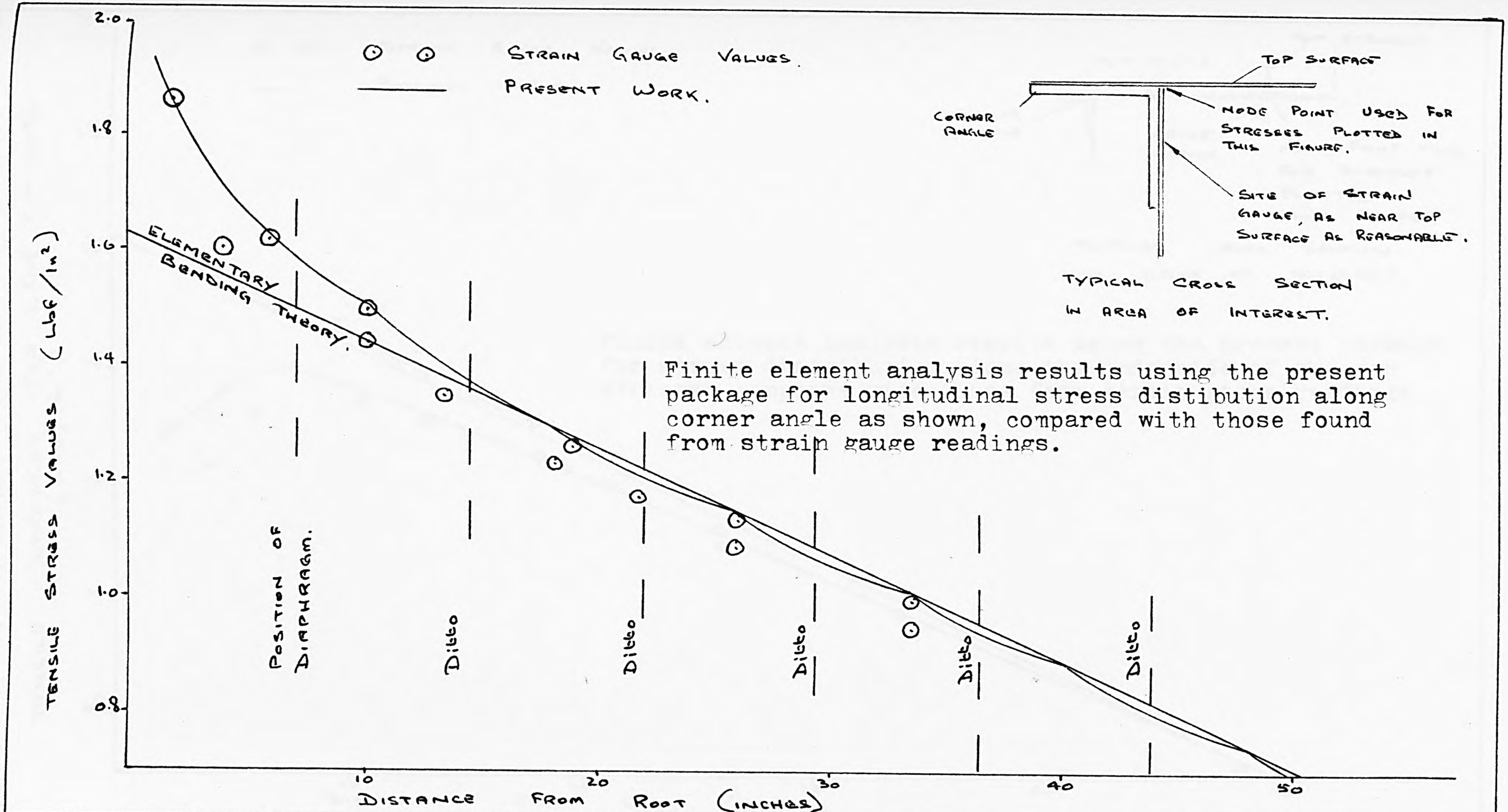


Restaints and load used in the box beam analysis of test 6.

Fig. Test 6.4



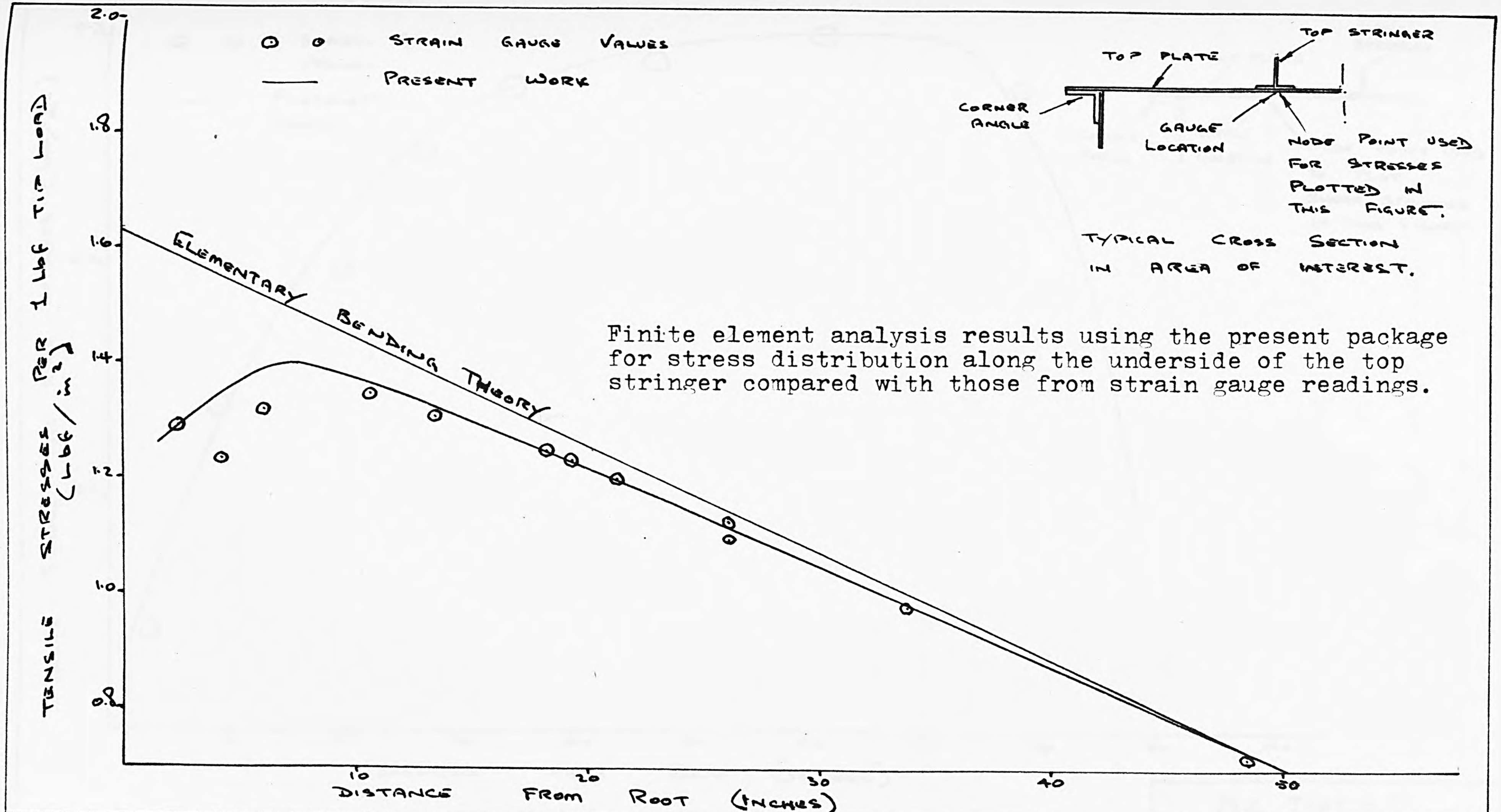
Longitudinal stresses on top surface of box beam as predicted by current work.	Fig. Test 6.5



Finite element analysis results using the present package for longitudinal stress distibution along corner angle as shown, compared with those found from strain gauge readings.

Curve number four of Fig. Test 6.5

Fig. Test 6.6



Finite element analysis results using the present package for stress distribution along the underside of the top stringer compared with those from strain gauge readings.

Curve number two of Fig. Test 6.5

Fig. Test 6.7

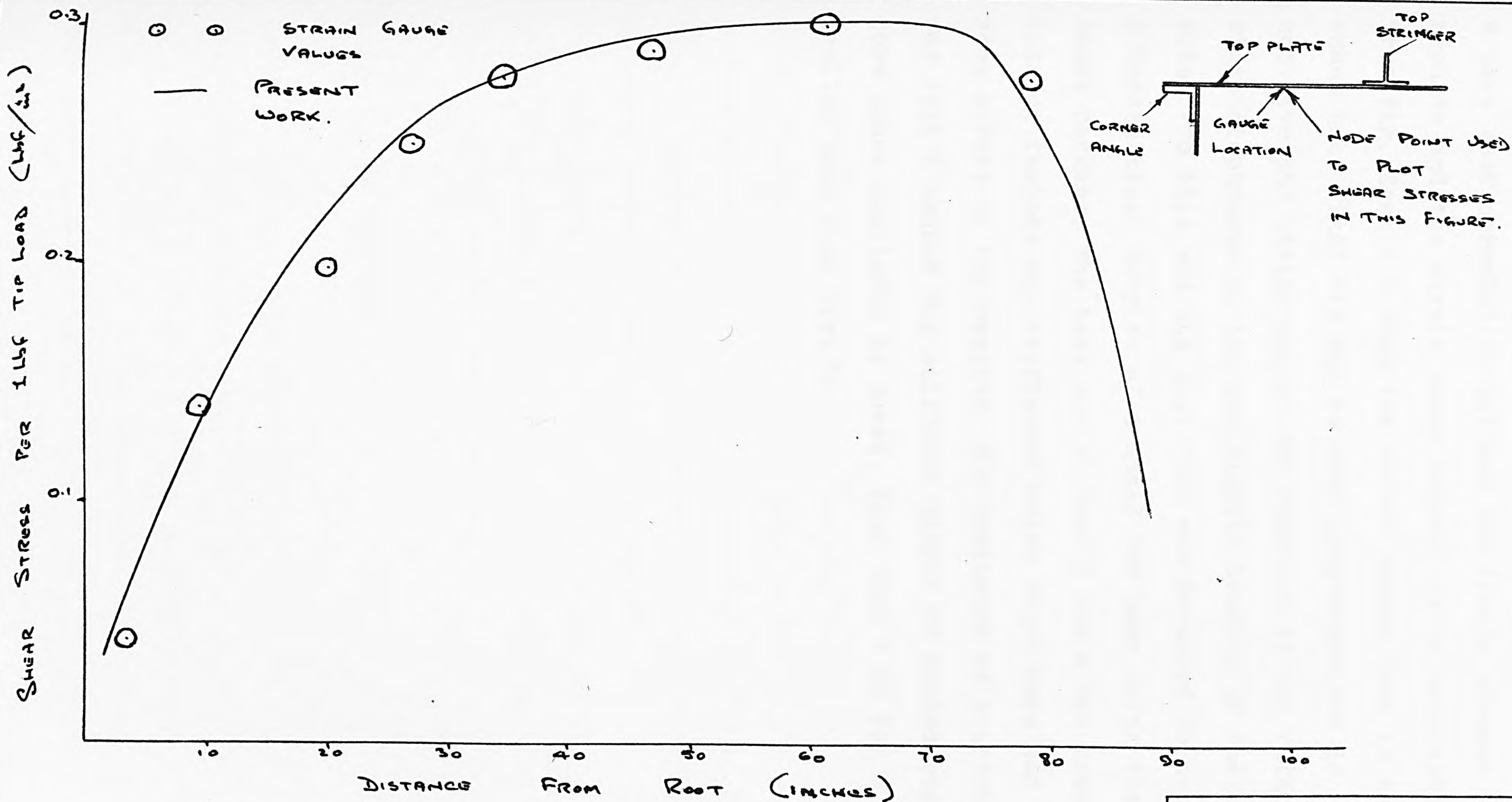


Fig. Test 6.8

Shear stress on top plate 10" from C.L. (Midway between top stringer and outer corner) a very good correlation between the finite element results and the strain gauge values. It is noticeable from Fig. Test 6.6 that the strain gauges have in most cases been fixed mid way between diaphragms and do not give any indication of the reaction of the relatively rigid diaphragms on the anticlastic bending of the sides. To this end the next test was arranged to analyse a hypothetical version of a small box beam using the Lusas package. The test run at Test 5 for a box beam did not include any stiffeners which might have had some effect on the results. The inclusion of stiffeners for test 5 caused the stiffness matrix to exceed the core space available to Lusas. Thus test 7 is for a smaller beam than test 5.

C.9 Test 7

C.9.1 Objective.

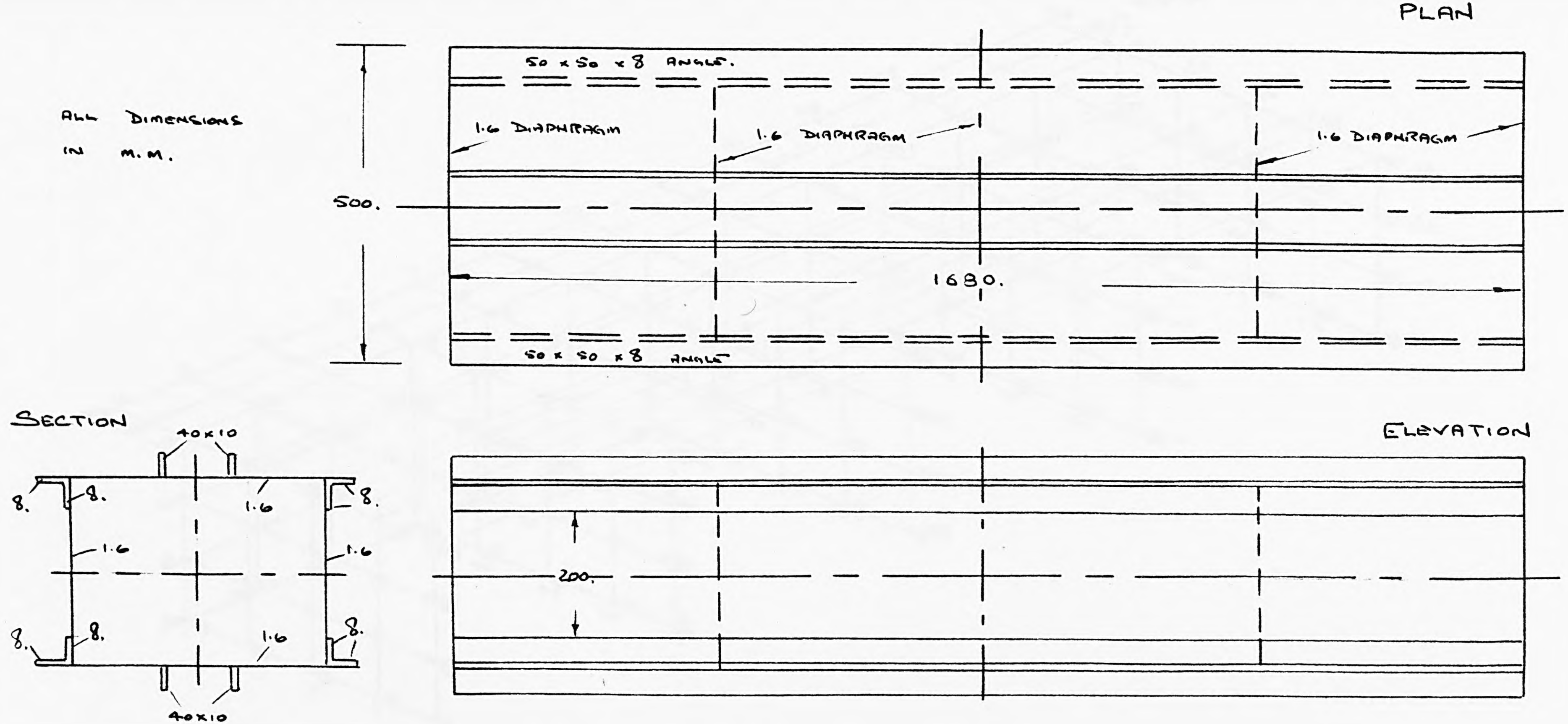
C.9.1.1 The strain gauge locations for the box beam of test 6 are positioned such that their results can be misinterpreted. A fair curve can be put through them to give the usually expected distribution.

C.9.1.2 The results of the analysis run as test 6 do not agree with this fair curve plot, though the analysis output still agrees with the strain gauge derived values.

C.9.1.3 This test 7 was run using Lusas to show, independently of the current package that a spiky set of curves was a justified interpretation of the strain gauges' results and that the output of the current package was reasonable and the fair curve normally fitted to the results was a mean value result. A spiky set of results has already been seen in the output of test 5 from both the current work and Lusas.

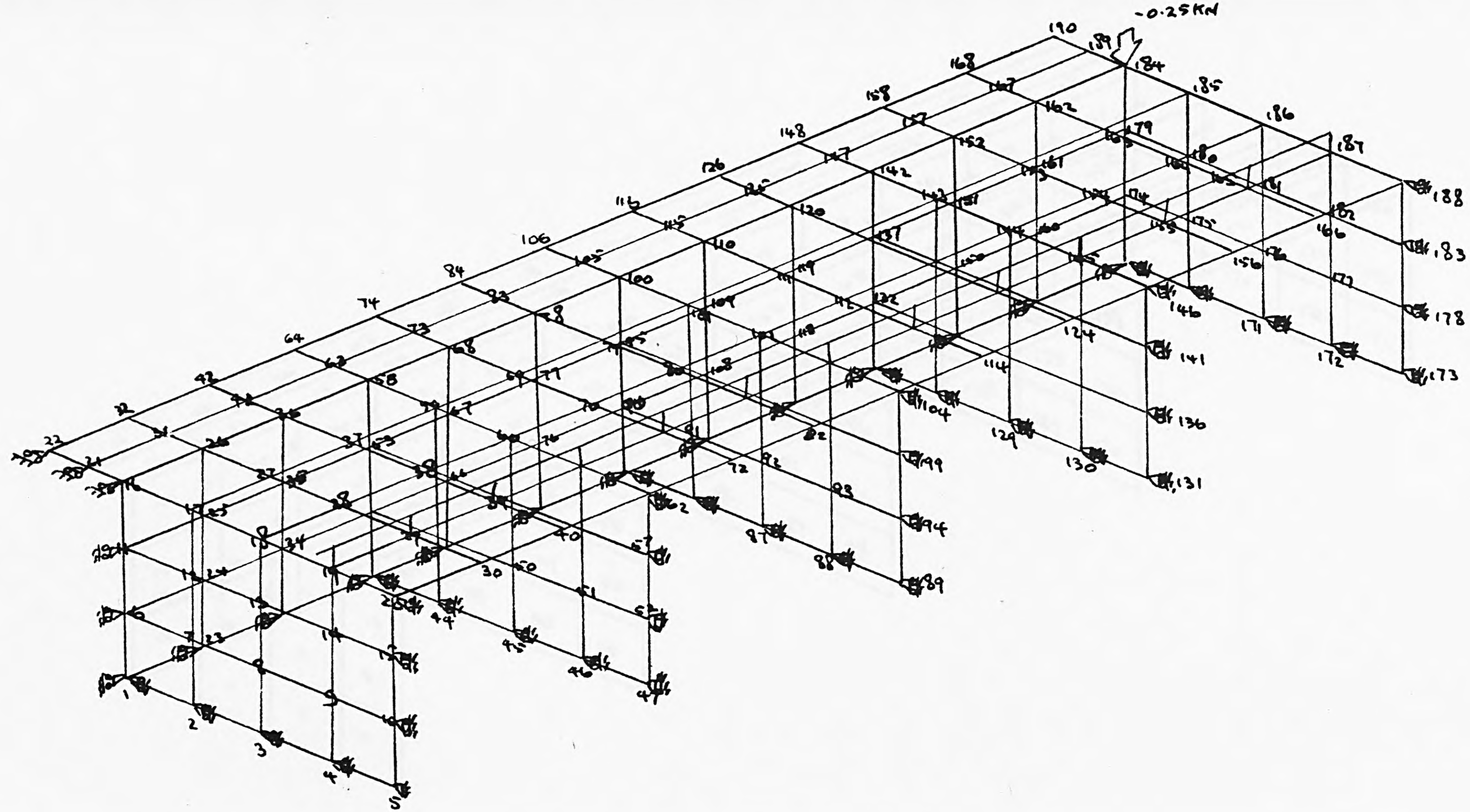
C.9.2 Description.

C.9.2.1 The structure used was a small version of the structure used for test 6. It was limited in size to the largest number of degrees of freedom which could be handled by Lusas. The difference between this test model and that used in test 5 is that test 5 contained no beam elements. The structural details



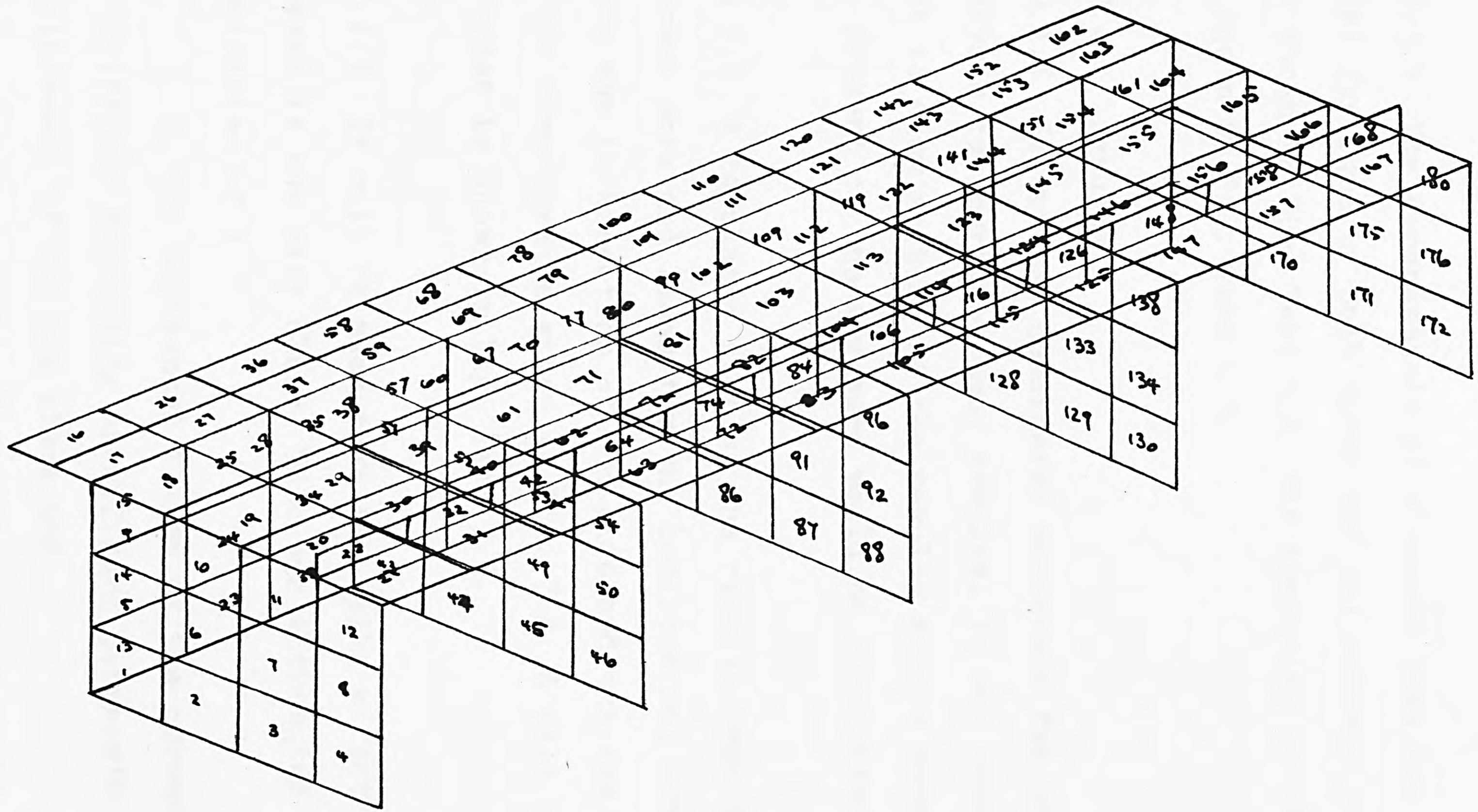
Plan elevation and section of full structure used in test number 7.

Fig. Test 7.1



Idealization and node numbering for Lusas run of test 7.

Fig. Test 7.2



Element numbering for Lusas run of test 7.

Fig. Test 7.3

for test 7 are shown in Fig. Test 7.1.

C.9.3 Idealization.

C.9.3.1 The arrangements of elements used for the model for runs by both Lusas and the current package are shown in Fig. Test 7.2. The numbering of elements is shown in Fig. Test 7.3.

C.9.4 Results.

C.9.4.1 It was not considered necessary for this analysis to reproduce many results. It was considered that to show spiking of the results when a beam element was present on the struture would be sufficient.

C.9.4.2 A tabulated list of the results from both the current work and Lusas for the longitudinal stresses along the intersection of the side and the top surface of the structure is given in Fig. 7.4. A plot of these stresses is shown in Fig. Test 7.5.

C.9.4.3 It will be seen that the spike effect though present is not very clear. This was attributed to a combination of :

a. The horizontal flange of the corner angle bar stiffener preventing significant poisson's ratio displacement of the top plate and

b. What little deflection which would occur in practice was not easily detected because there were insufficient plate elements. There number being restricted because of the limited capacity of Lusas.

DISTANCE FROM MID LENGTH (M.M.)	STRESS FROM CURRENT PACKAGE KN/MM²	STRESS DIFFERENCES PER M.M	STRESS FROM LUSAS KN/MM²	STRESS DIFFERENCES PER M.M
35	1.48 E-3		1.380 E-3	
		5. E-6		4.357E-6
105	1.13 E-3		1.075E-3	
		1.857 E-6		1.871E-6
175	1.00 E-3		0.944E-3	
		1.800 E-6		1.800E-6
245	0.874 E-3		0.818E-3	
		1.786 E-6		1.757 E-6
315	0.749 E-3		0.695 E-3	
		1.529 E-6		1.600 E-6
385	0.642E-3		0.583E-3	
		1.471 E-6		1.514 E-6
455	0.539E-3		0.477E-3	
		1.486E-6		1.314 E-6
525	0.435E-3		0.385E-3	
		1.443E-6		0.914E-6
595	0.334E-3		0.320E-3	
		1.286E-6		0.586E-6
665	0.244E-3		0.282E-3	

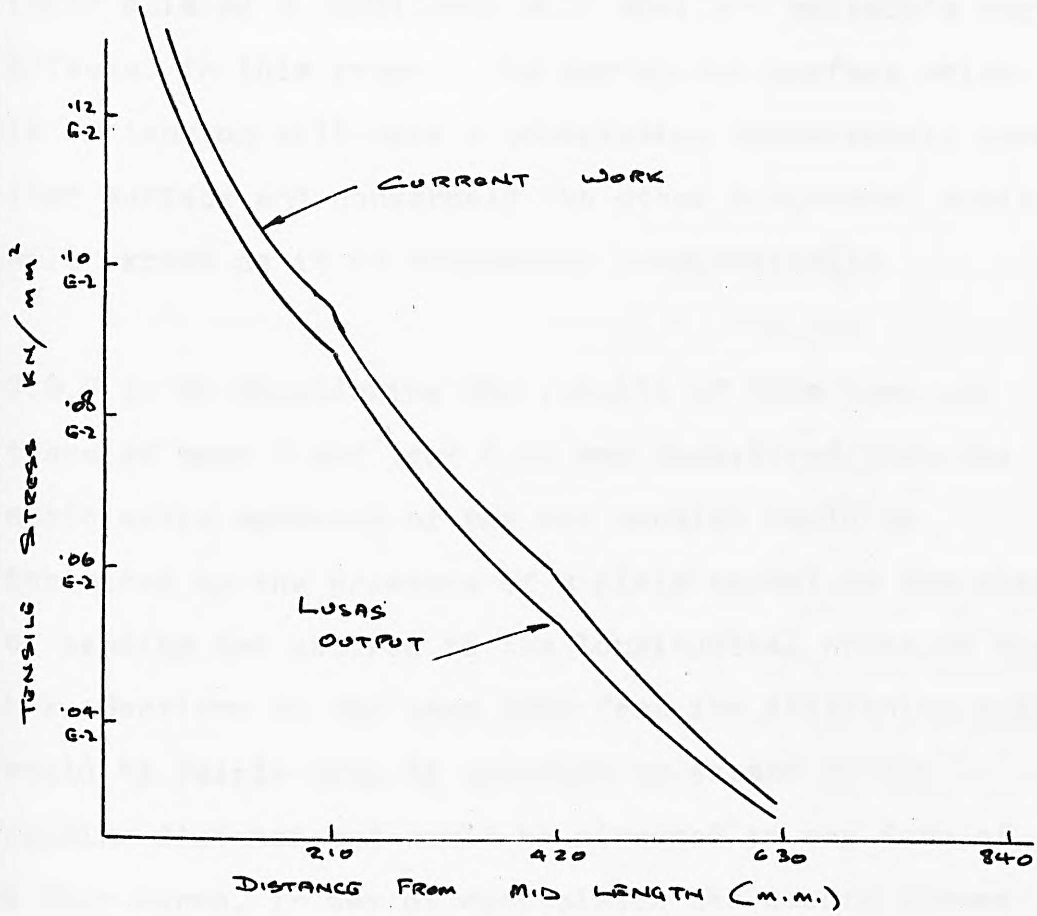
Table of stresses produced from current package and Lusas and columns of differences.

Fig. Test 7.4

C.9.4.4 To show up the positions of spikes or disturbances in an otherwise smooth curve, a plot was made of the difference column. This is shown in Fig. Test 7.6 and it is clear that in way of the beam's internal stiffening plates, the difference plot tends to go horizontal, which can for this model be seen more easily than the spikes.

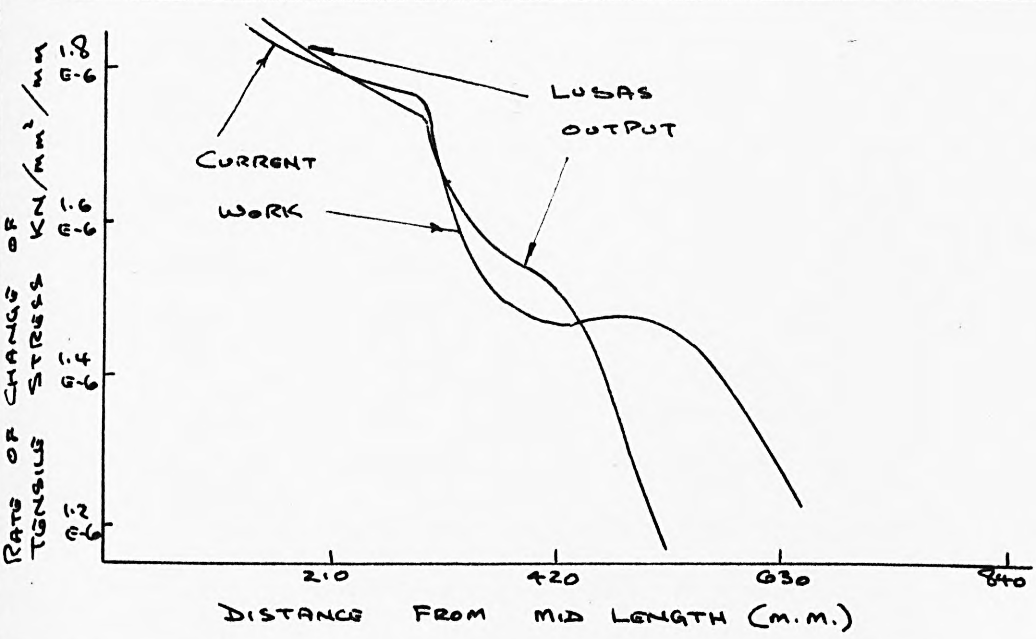
C.9.5 Conclusions.

C.9.5.1 It is well known that a box beam when acting



Longitudinal stress distibution along the intersection of the side and top of the structure in Fig. Test 7.1

Fig. Test 7.5



Rate of change of stresses shown in Fig. Test 7.5

Fig. Test 7.6

inter alia as a cantilever will manifest poisson's ratio effects. In this respect the horizontal surface which is in tension will have a contraction transversely across that surface and conversely the other horizontal surface will expand as it is compressed longitudinally.

C.9.5.2 In considering the results of this test and those of test 5 and test 6 it was considered that the anticlastic movement of the box section would be inhibited by the presence of a plate normal to the plane of bending and secured to the longitudinal sides of the box. Sections of the beam away from the stiffening plates would be fairly free to contract or expand as the bending dictated and would be stressed in the form of a fair curve. In way of such plates the smooth stress pattern would be upset by the considerable resistance to distortion in their own planes.

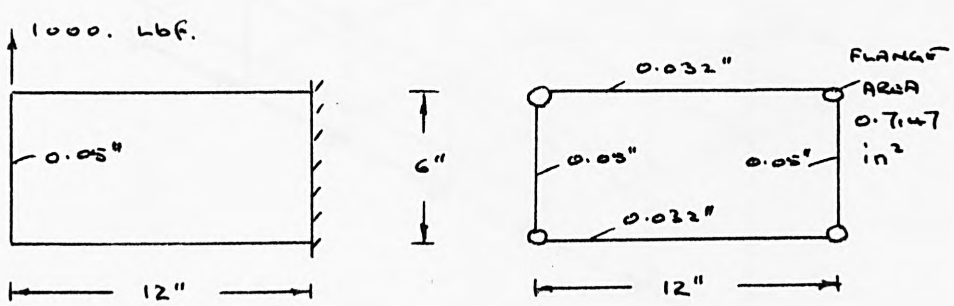
C.10 Test 8

C.10.1 Objectives.

C.10.1.1 When this work was commenced it was anticipated that there would be no difficulty in obtaining results of the structural analysis of large structures. An examination of the results available for inspection shows that usually full reports are not available and that in almost every case some of the input data is missing.

C.10.1.2 One case where the data used for an analysis was available together with the results and an independent check was contained in the thesis of Tarpy(61). Though this thesis was dealing with the anaysis of large structures, the test case was quite small.

C.10.2 Description.



Section of aluminium box used by Tarpy(61) for his analysis of a box beam.

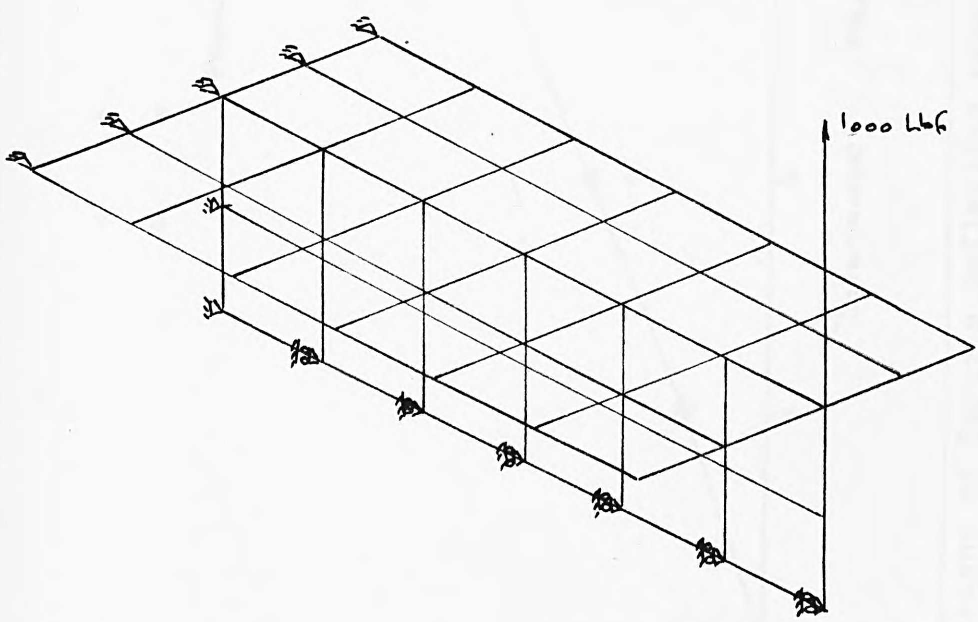
Fig. Test 8.1

C.10.2.1 An aluminium sheet metal box shown in Fig. Test 8.1 was subjected to a 1000 lbf static load at each of the upper corners at the free end.

C.10.3 Idealization.

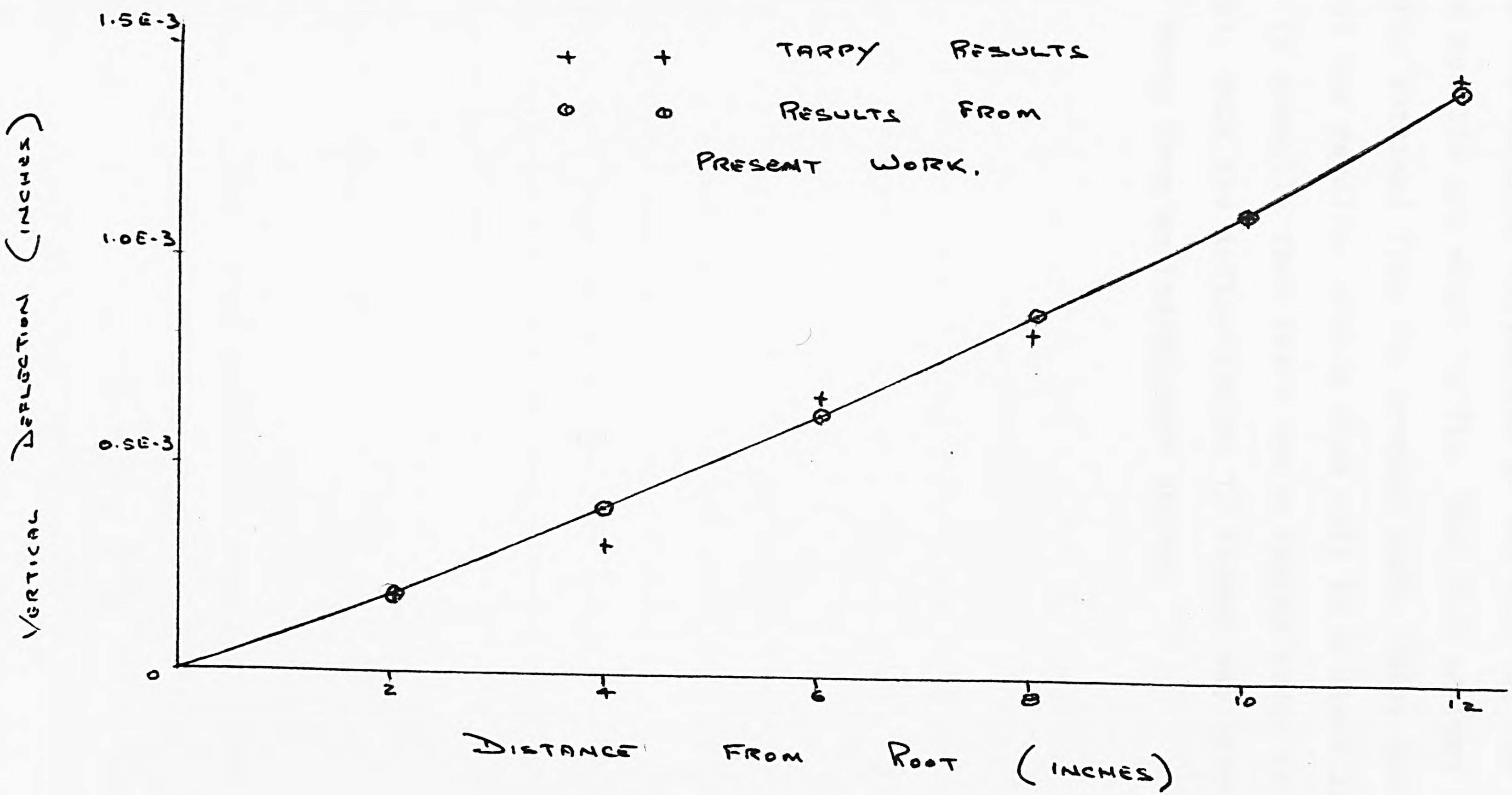
C.10.3.1 The Tarpy idealization used triangular plate elements carrying only shear for the sheet material with the flanges represented by elements carrying direct stress only.

C.10 3.2 The analysis by the present work used one rectangular element for every two triangular elements used by Tarpy. The flanges were represented by plate elements.



Ideaization of the Tarpy box beam for analysis by the present work.

Fig. Test 8.2



Comparison of Tarpy's deflection results and the present work for Test 8
The Tarpy results contained no stress values.

Fig. Test 8.3

C.10.5 Results.

C.10.5.1 The only results given by Tarpy are those for the vertical deflection of the side of the box. His results are shown in Fig. Test 8.3 against the curve obtained from the present work. Tarpy did not plot his results, giving them only in a short list. It is possible that there was a typing error in the list. Only the deflection at 12 inches was confirmed by Tarpy from an independent source.

C.11 Test 9.

C.11.1 Objective.

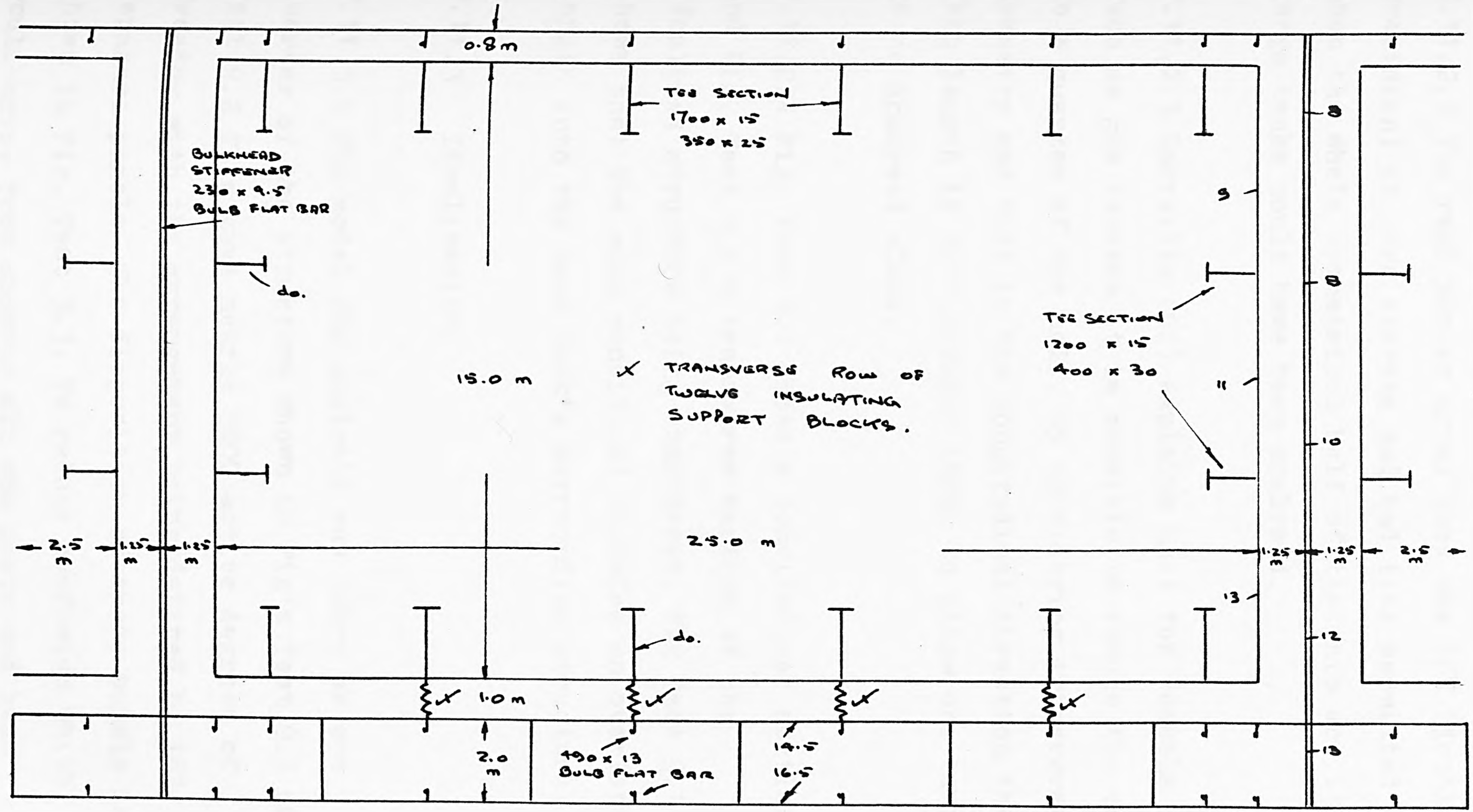
C.11.1.1 To show how a gas tanker could be structurally analysed using the techniques already described for the iterative process in removing thermal insulating supports from the analysis without resort to the current practice of manually deleting the contribution of the supports and then re-running the whole analysis.

C.11.1.2 To improve on the results achieved by the usual methods at present available in this field by improving the method of restraining body movements of the gas tank structure.

C.11.1.3 Loads representing hydrostatic forces due to both sea water and liquefied gas cargoes, inertia from surge, heave and roll, thermal, shearing and bending forces were used to show that the package could handle them though their detailed evaluation was considered outside this work.

C.11.2 Description.

C.11.2.1 A limitation applied by the City University on the computer run time prevented a whole ship and its full set of cargo tanks from being analysed and in this respect this particular analysis does therefore not vary from established practice except that the



C.L. section of zone of tank and adjacent structure used for test 9.	Fig. Test 9.1

time used was much less than established practice.

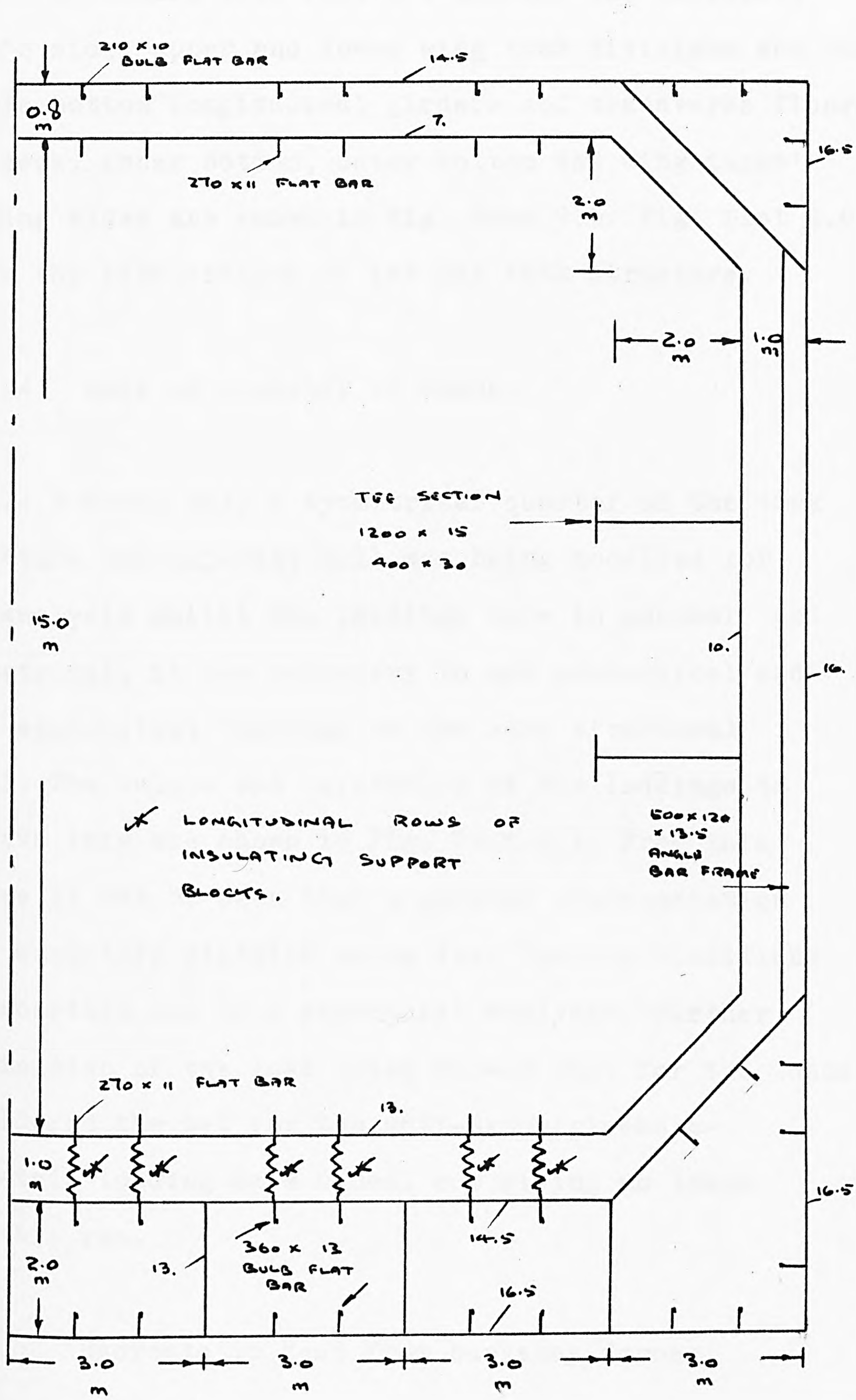
C.11.2.2 The real number array used was 56K (double precision) of core storage and had time permitted then the whole symmetrical half of the ship and its cargo tanks could have been analysed.

C.11.2.3 Latreille (62) explains that for vessels such as gas tankers it is possible to reduce the model to a quarter of one hold, by considering transverse symmetry and that in the longitudinal direction the ship length is sufficiently large to allow one tank to be analysed alone.

C.11.2.4 Fig. Test 9.1 shows a longitudinal section and Fig. Test 9.2 a transverse section of the idealized structure being considered. Fig Test 9.1 shows that the zone considered includes an overlapping length into the next tank's surrounding structure.

C.11.3 Idealization.

C.11.3.1 The model for analysis was taken as one quarter of the structure shown in Fig's Test 9.1 and Test 9.2 This gave nearly 5000 active degrees of freedom with the arrangement being defined by 187 separate panels. The disposition of these panels is shown in Fig. Test 9.3. To reduce confusion which would arise from showing all the plate and beam elements on one drawing, three drawings are used.



Transverse section of tank and adjacent structure used for test 9.

Fig. Test 9.2

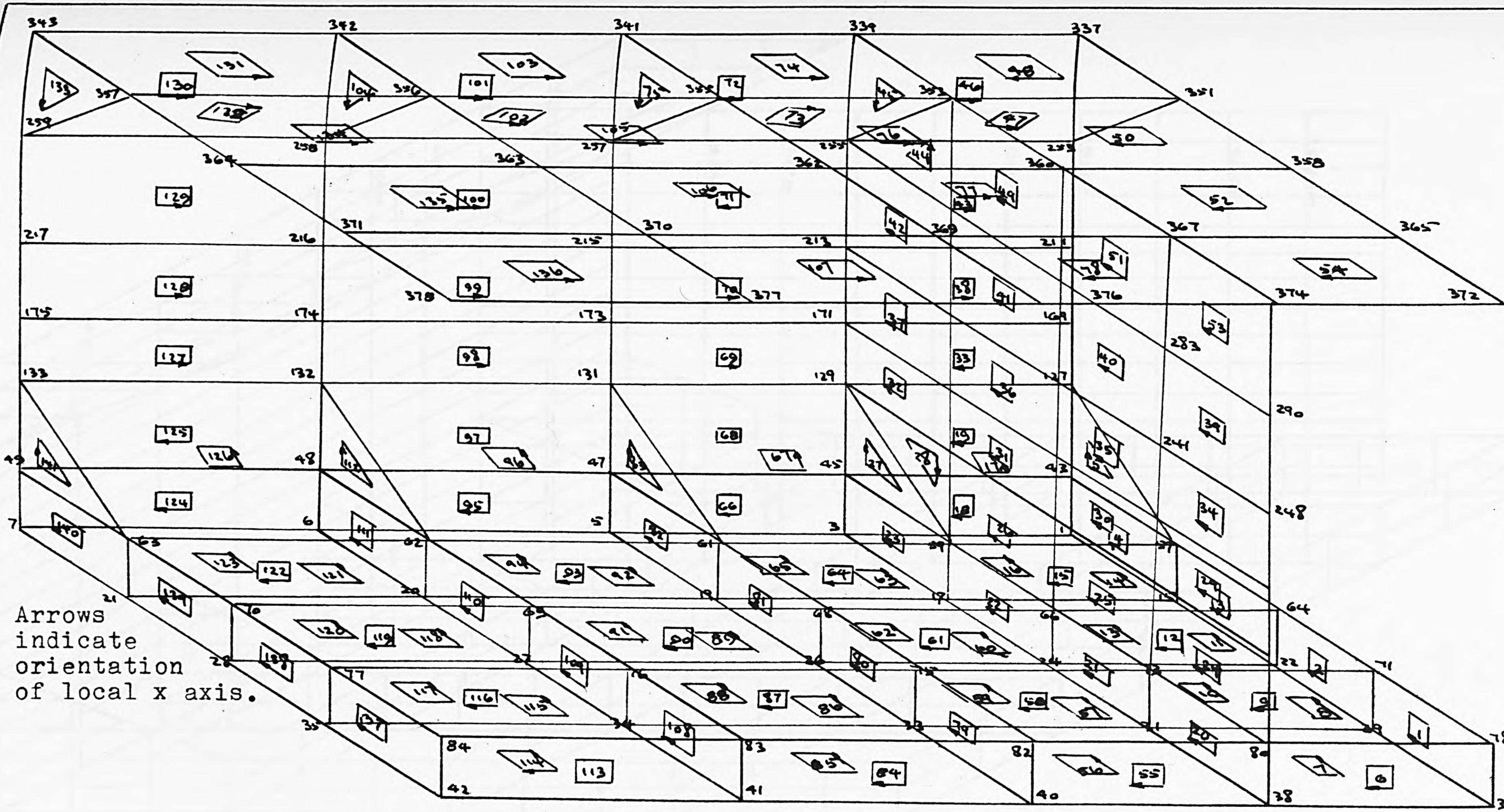
Fig's Test 9.4 and Test 9.5 show the idealization of the ship's structure with Test 9.4 showing the bulkhead, ship's side, upper and lower wing tank divisions and the double bottom longitudinal girders and transverse floors. The deck, inner bottom, outer bottom and wing tanks' sloping sides are shown in Fig. Test 9.5. Fig. Test 9.6 shows the idealization of the gas tank structure.

C.11.4 Lack of symmetry of loads.

C.11.4.1 Since only a symmetrical quarter of the tank structure and adjacent hull was being modelled for the analysis whilst the loadings were in general not symmetrical, it was necessary to use symmetrical and anti symmetrical loadings on the same structural model. The values and derivation of the loadings to achieve this are shown in Fig. Test 9.7. From this figure it can be seen that a quarter representation of a structure dictates using four loading conditions to ascertain the full structural analysis. Further examination of the load cases showed that for the loads considered the set for the anti-symmetric-anti-symmetric loading case cancel out giving no loads for this run.

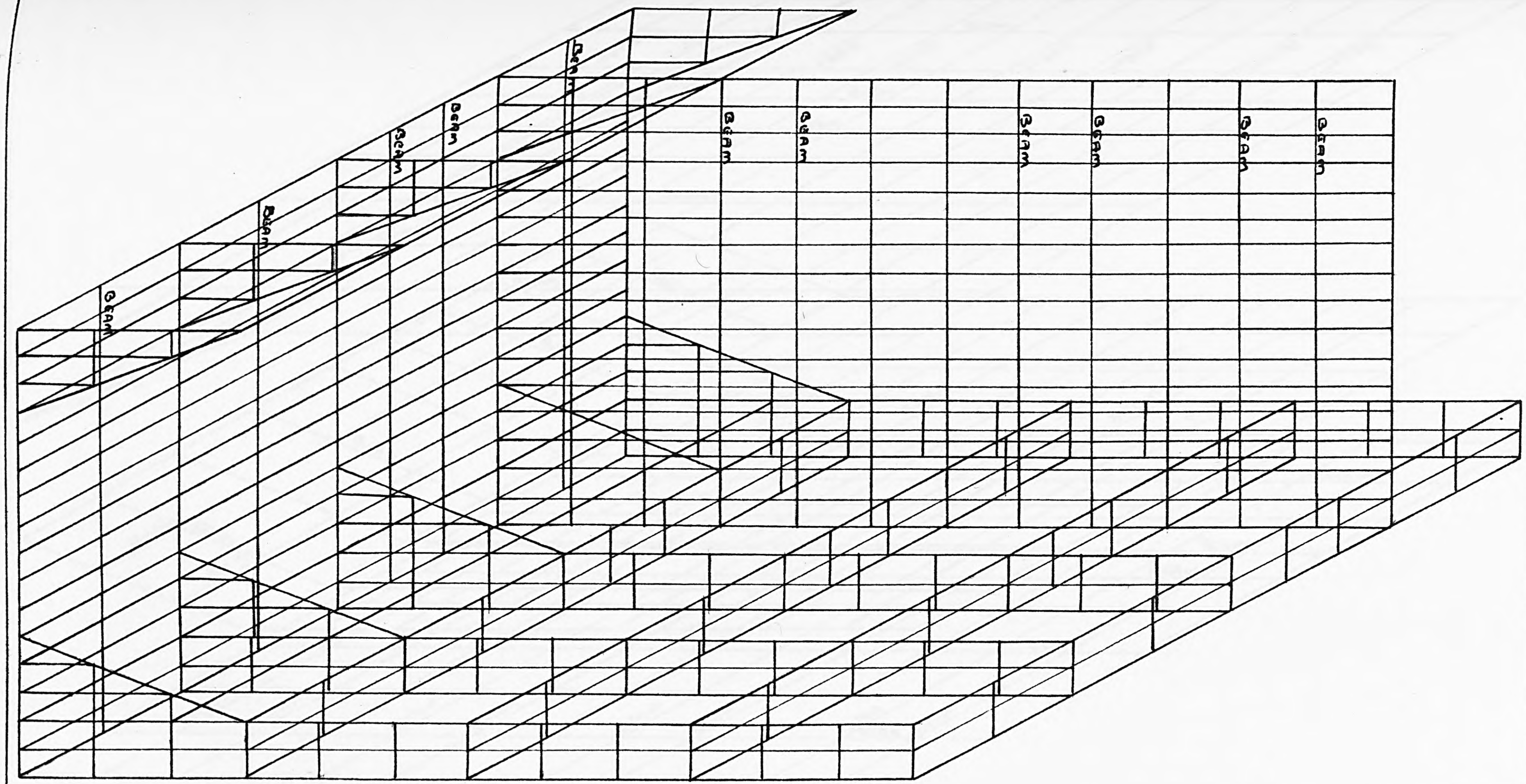
C.11.5 Hydrostatic head from buoyancy forces.

C.11.5.1 The vessel was considered to be at an angle of heel such that the draughts on the port and starboard sides were respectively 8m and 4m. The results to be

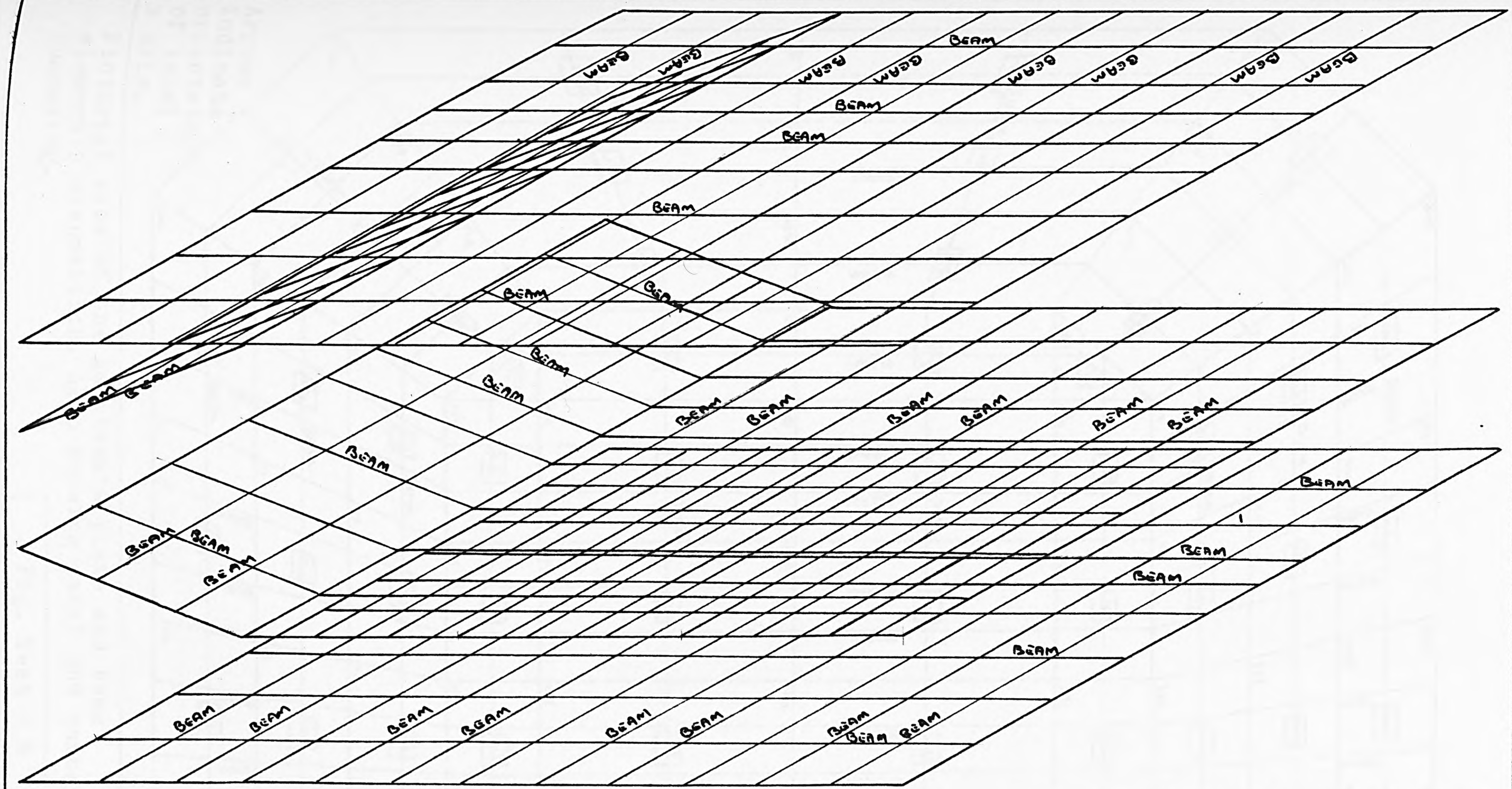


Panel and corner numbering of hull model used in test 9.

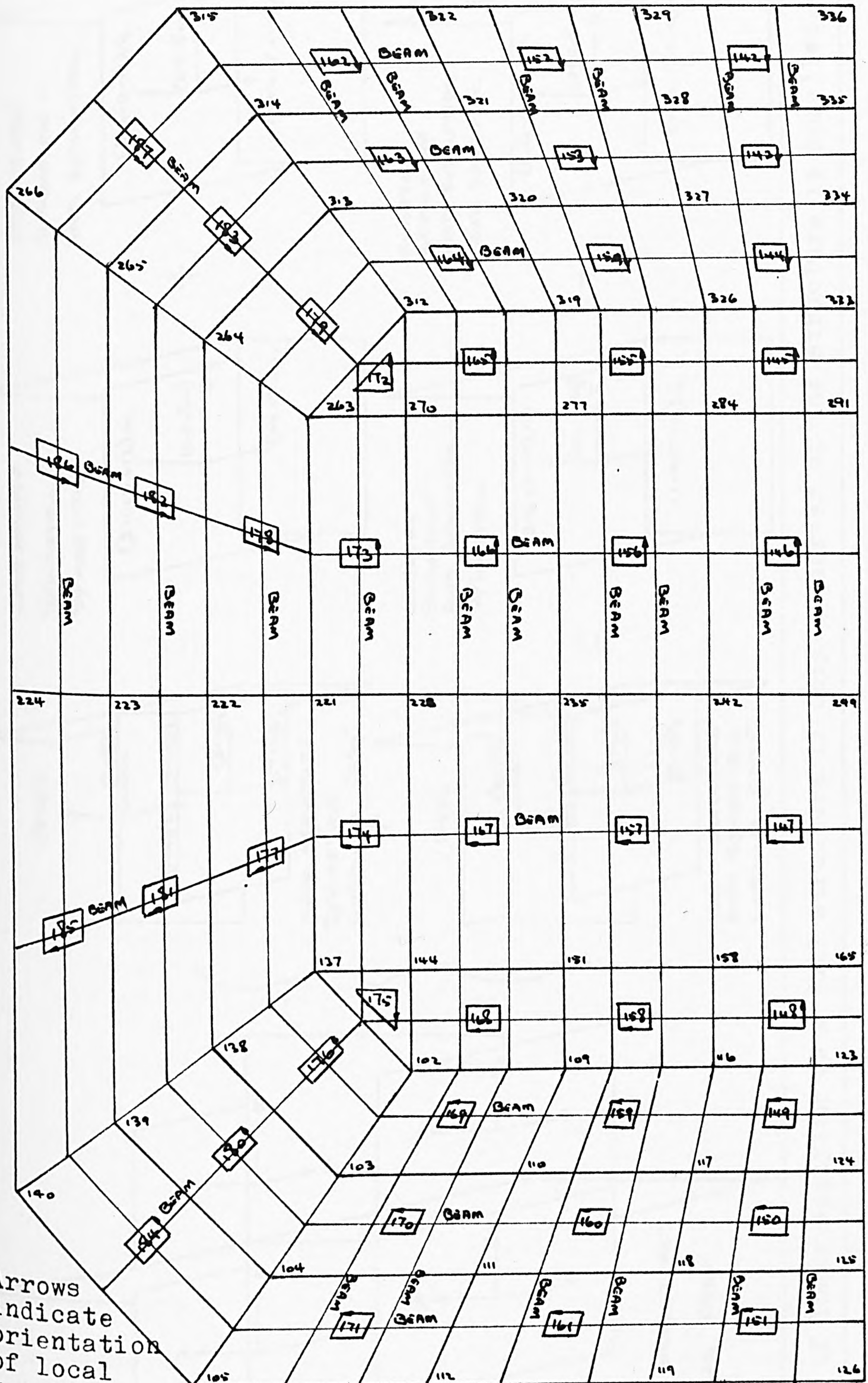
Fig. Test 9.3



Arrangement of plate and beam elements for vertical surfaces used in test 9. | Fig. Test 9.4



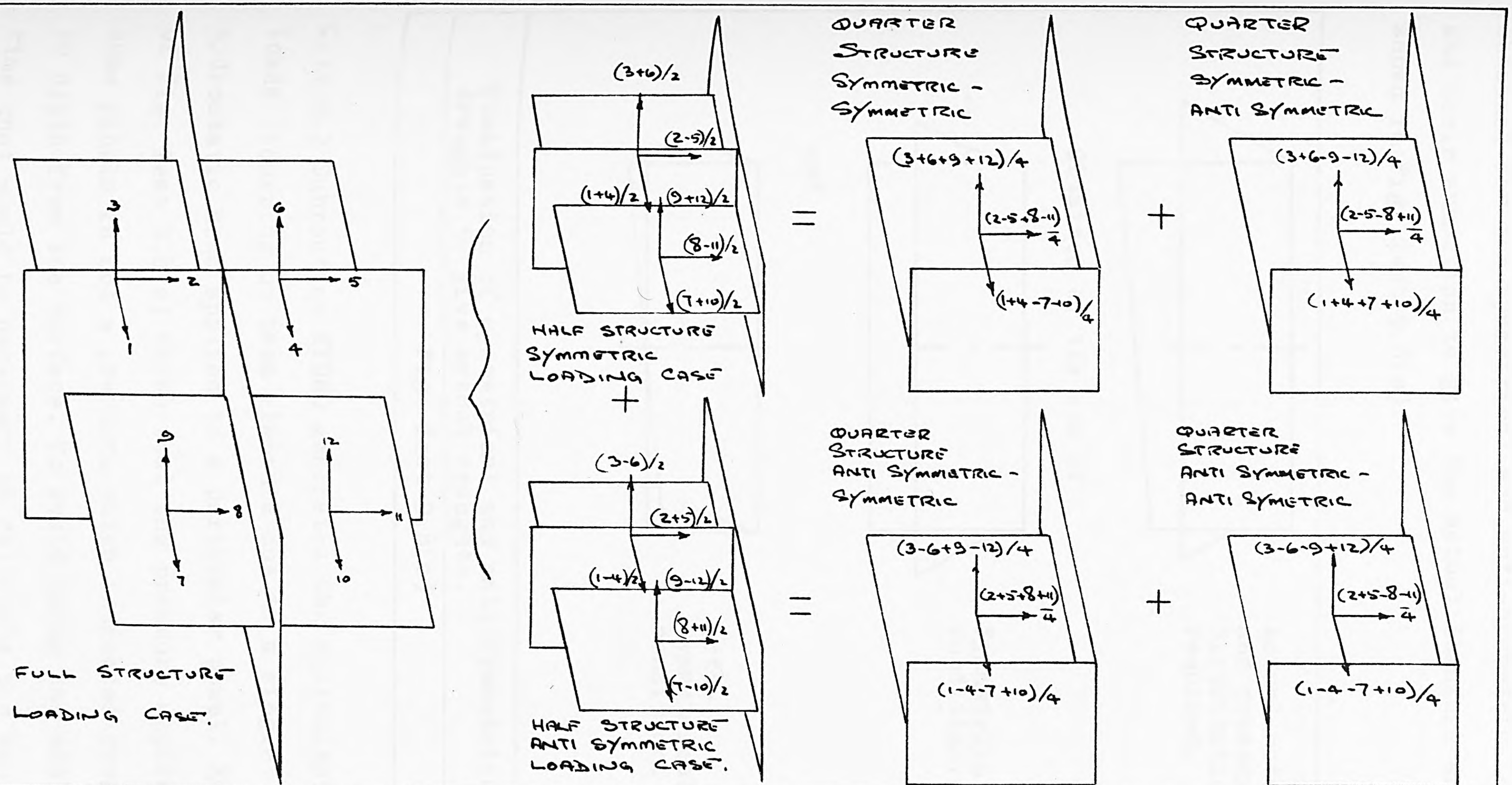
Arrangements of plate and beam elements for non vertical surfaces used in test 9. | Fig. Test 9.5



Arrows indicate orientation of local x axis.

Pictorial view of the gas tank's plate and beam element's disposition. Also showing panel and corner numbering.

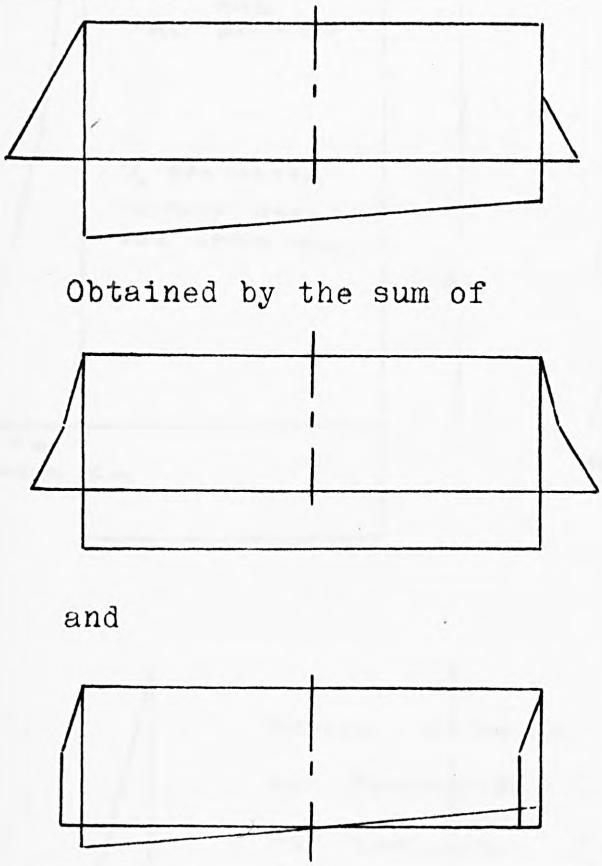
Fig. Test 9.6



Treatment of loads applying to a full structure if only a quarter of the structure is modelled.

Fig. Test 9.7

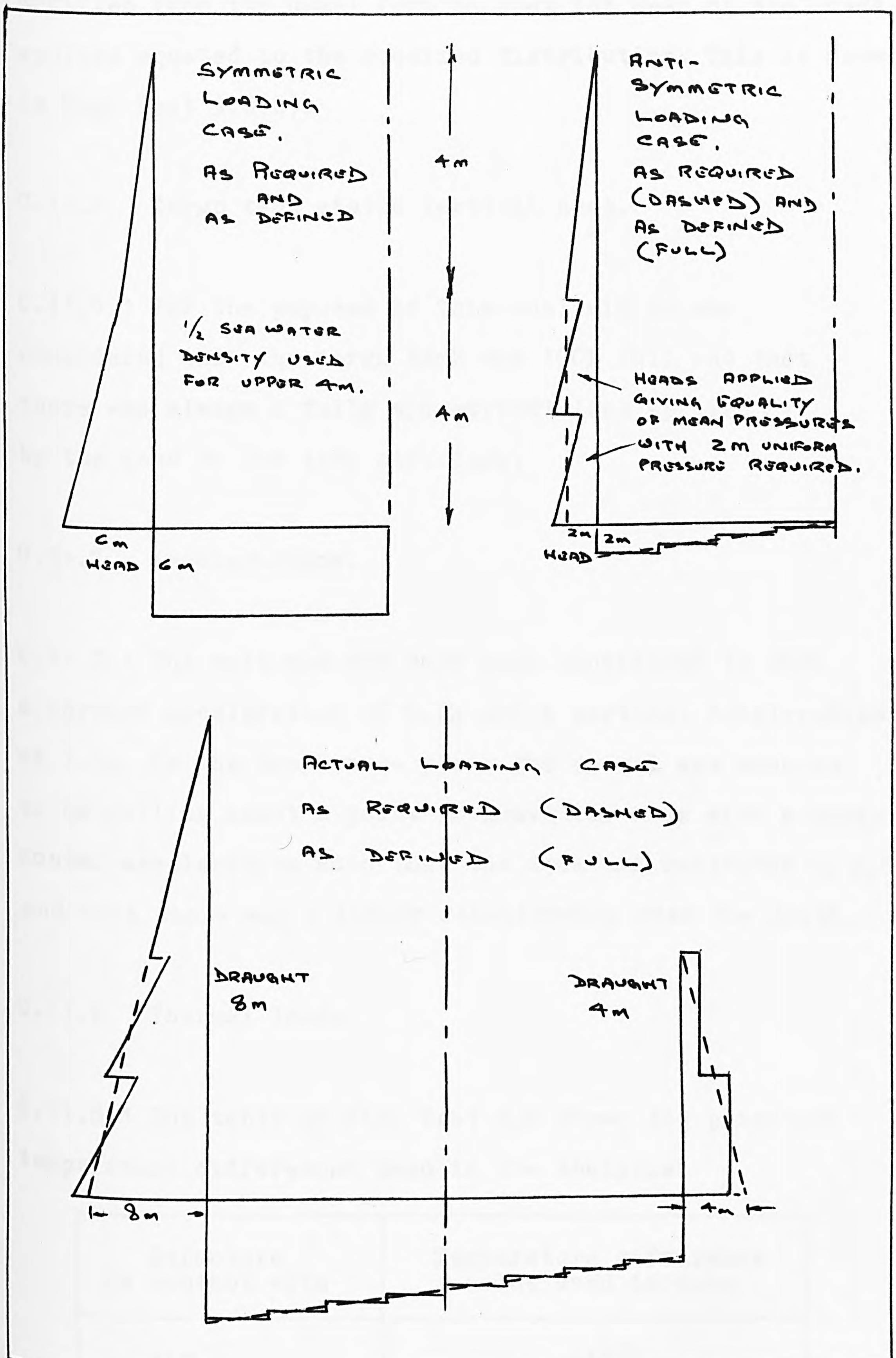
obtained for the symmetrical and anti-symmetric loadings and their summation to give the actual loadings are shown in Fig. Test 9.8(a).



Combination of symmetrical and anti-symmetrical draughts to give actual draughts.

Fig. Test 9.8(a)

C.11.5.2 Subroutine HYDRO generates the equivalent nodal loads occurring on beam elements due to a single hydrostatic head applied to a particular panel. Examination of Fig. Test 9.8(a) shows that the pressure applied to some panels is not a pressure which increases proportionally to depth from the surface. To avoid using the additional time that would be necessary to calculate the equivalent point loads by hand, the data supplied for HYDRO was



Combination of symmetric and anti-symmetric loads required to give data for a full analysis using a quarter structure model with asymmetrical draughts. Showing the difference between actually required profile and the package generated profile.

Fig. Test 9.8(b)

modified from its usual form so that the mean of the loads applied equated to the required distribution. This is shown in Fig. Test 9.8(b).

C.11.6 Cargo tank static vertical head.

C.11.6.1 For the puposes of this analysis it was considered that the cargo tank was 100% full and that there was always a fully symmetrical load applied by the head to the tank structure.

C.11.7 Accelerations.

C.11.7.1 The hull and the tank were considered to have a forward acceleration of 0.2g and a vertical acceleration of 1.5g. In the transverse plane the vessel was assumed to be rolling about a point 5m above the base with a horizontal acceleration such that the deck was subjected to 0.5g and that there was a linear relationship over the depth.

C.11.8 Thermal loads.

C.11.8.1 The table at Fig. Test 9.9 shows the principal temperature differences used in the analysis.

Structure in contact with	Temperature difference value used in data
Air	$-18^{o}C$
Sea water	$0^{o}C$
Cargo	$-48^{o}C$

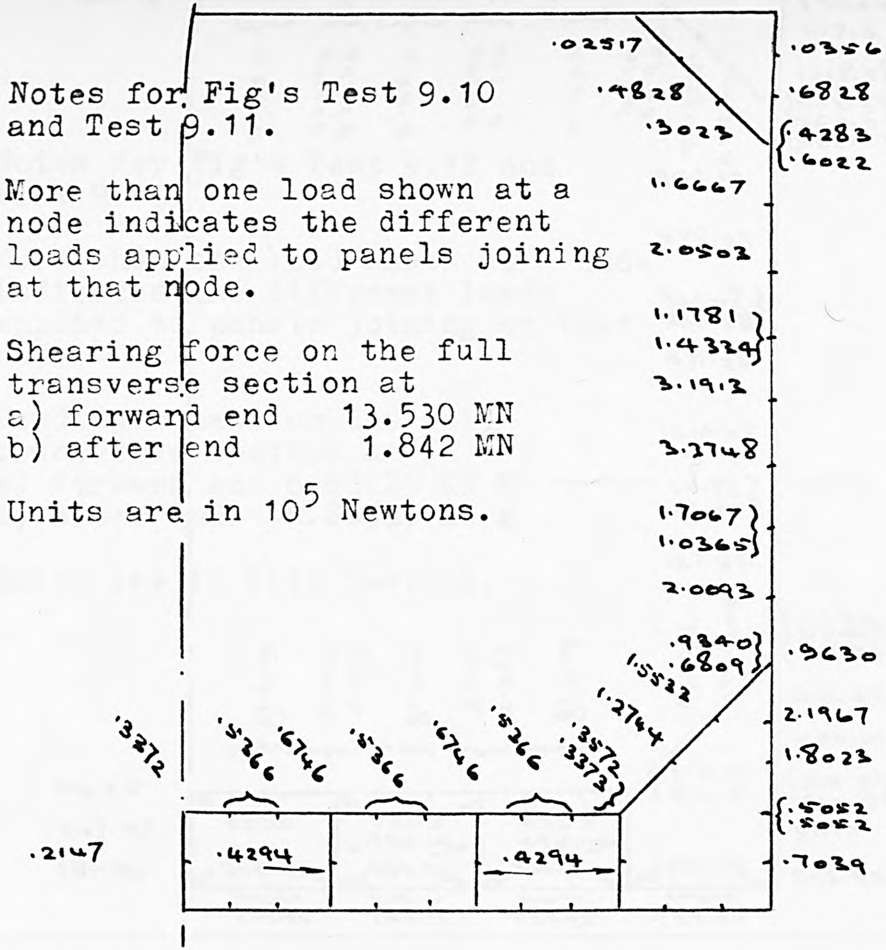
Temperature Table — Fig. Test 9.9

Notes for Fig's Test 9.10 and Test 9.11.

More than one load shown at a node indicates the different loads applied to panels joining at that node.

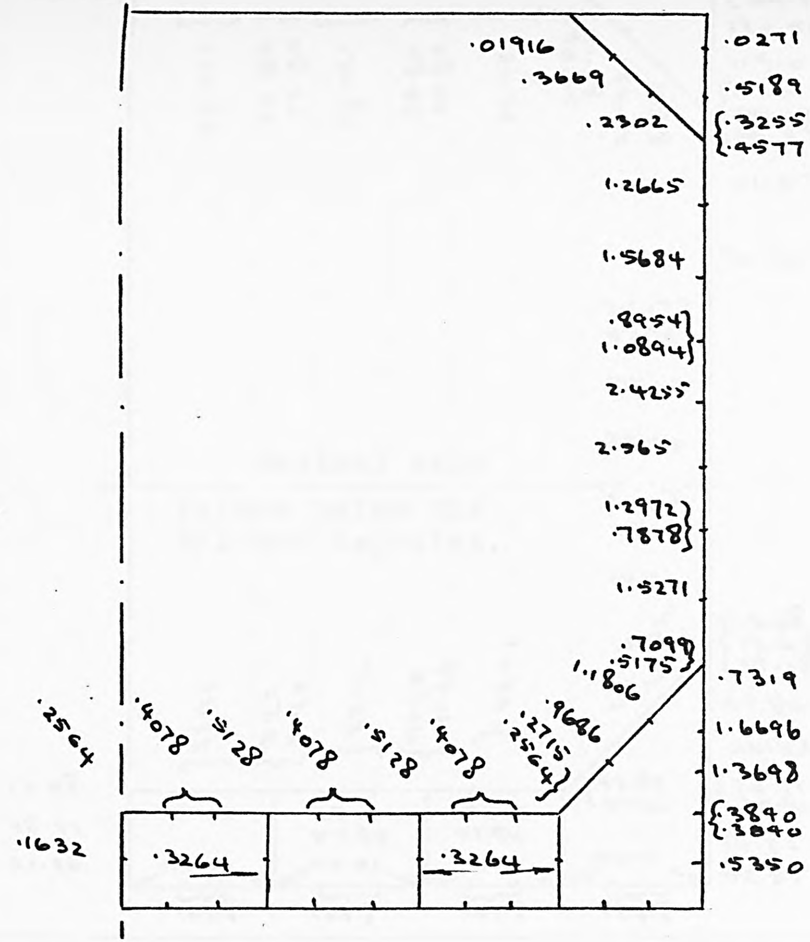
Shearing force on the full transverse section at
a) forward end 13.530 MN
b) after end 1.842 MN

Units are in 10^5 Newtons.



Nodal loadings representing assumed boundary shearing forces used in the symmetrical-symmetrical condition.

Fig. Test 9.10



Nodal loading representing assumed boundary shearing forces used in the anti-symmetrical-symmetrical condition.

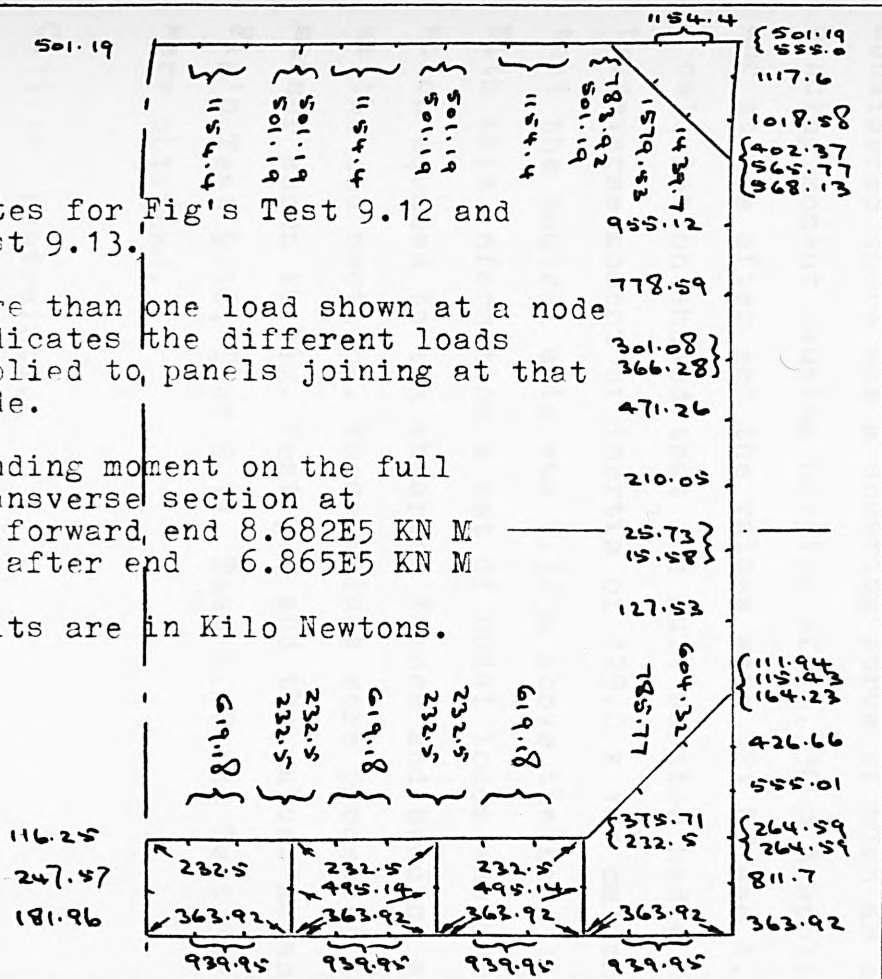
Fig. Test 9.11.

Notes for Fig's Test 9.12 and Test 9.13.

More than one load shown at a node indicates the different loads applied to panels joining at that node.

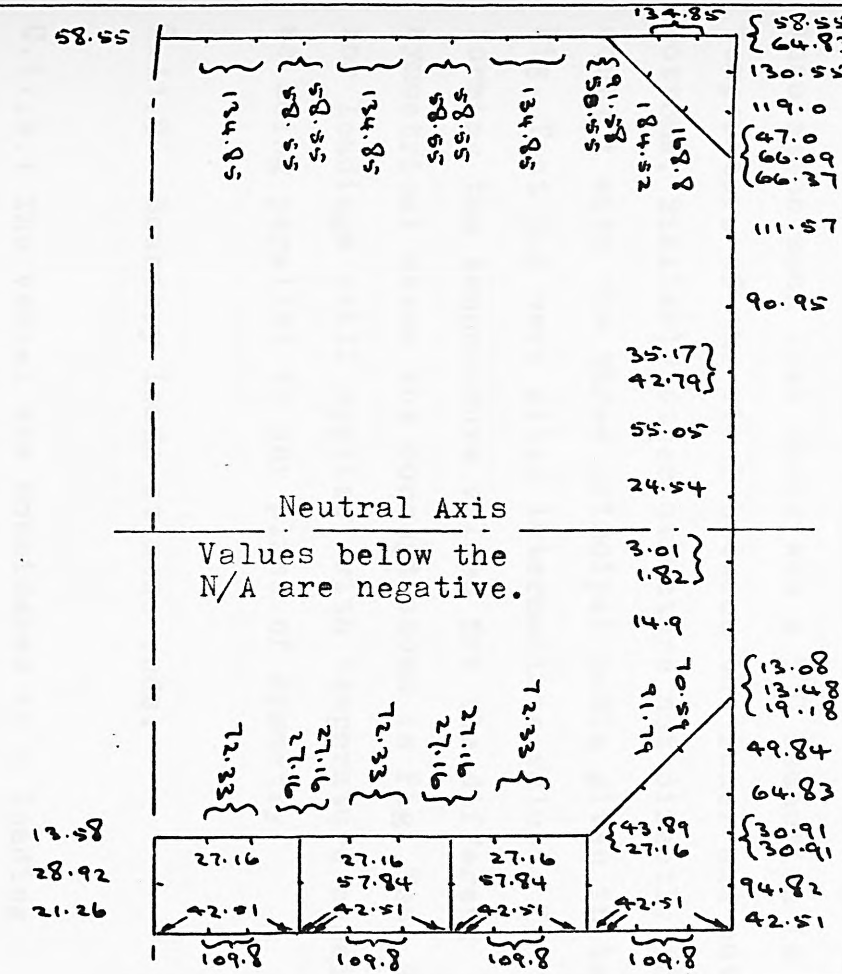
Bending moment on the full transverse section at
a) forward end 8.682E5 KN M
b) after end 6.865E5 KN M

Units are in Kilo Newtons.



Nodal loadings representing assumed boundary hogging bending moments used in the symmetrical-symmetrical condition.

Fig. Test 9.12



Nodal loadings representing assumed boundary hogging bending moments used in the anti-symmetrical-symmetrical condition.

Fig. Test 9.13

Some thermal flow was allowed for through the insulation such that there was a $5^{o}C$ reduction in temperature of the steel between the inner and outer bottoms. Similarly other structure not directly in contact with the three principal media given in table Fig. Test 9.9 were given intermediate values. In forming the temperature values for the different symmetrical cases the concept shown in Fig. Test 9.7 for loadings still applies, with temperature considered as being parallel to any plane of symmetry.

C.11.9 Boundary loads at zone ends.

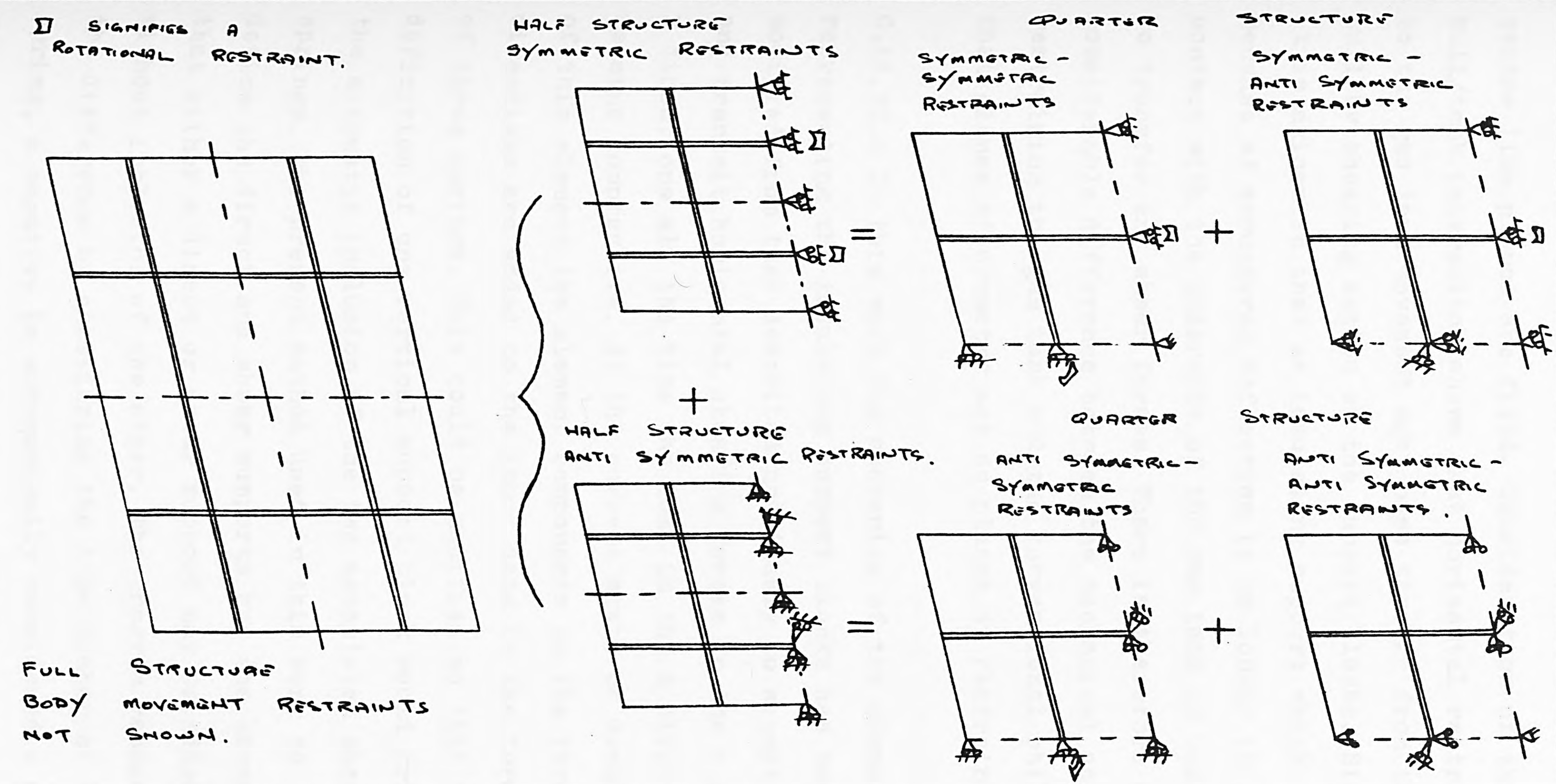
C.11.9.1 The vessel was considered in a loading condition such that at the forward end of the zone considered there was a shearing force of 6765 KN and a bending moment causing hogging of 13.530 m N applied and at the after end the values were 921 KN and 1.842 m N. A calculation showed that the hull section had a transverse moment of inertia of 129.0 x 10^3 cm^4 and that the neutral axis was 7.323m above the base line. With this information a set of nodal loads was derived which equated to the shearing forces and bending stresses at the end sections. These values were processed in the manner shown in Fig. Test 9.7 and the values shown in Fig's Test 9.10, Test 9.11, Test 9.12 and Test 9.13 were obtained.

C.11.10 Restraints.

C.11.10.1 As explained in the section dealing with loads for this test case, only a quarter of the structure is modelled requiring the basic problem to be solved more than once. To achieve an analysis of the full structure whilst using only a symmetrical part of the structure requires the use of restraints along the edges of symmetry of such an alignment that when the various solutions are selectively added an analysis of the full structure results. Fig. Test 9.14 shows the restraints to be used with the freedoms available in this work for the derivation from the full structure to the symmetrical quarter.

C.11.10.2 In addition to restraints necessary to run a symmetrical part of a whole structure, stuctural restraints are also necessary to stop body movements. In the case of the model representing the gas tank, body movement restaint is provided from the hull structure through the support blocks. For the hull structure the middle line longitudinal and transverse planes are used as planes of symmetry and the vertical movement is absorbed by a vertical restraint set in the transverse frame nearest the centre of the gas tank's bottom.

C.11.10.3 An examination of the relevant literature shows that the current method of analysing this type of structure ignores the lack of definite body restraints to the gas tank structure in its horizontal plane. This is currently overcome by assuming that the vertical



Restraints applied to a quarter structure so that the results may be summed give those for the full structure.

Fig. Test 9.14

centre line planes are fixed. Consideration of the hull/tank interaction shows that horizontal restraint to the gas tank movement must pass through from the hull by shearing action of the support blocks. Since it is axiomatic that an insulating support which because of structural deflection is no longer in contact with the underside of the gas tank is unable to transfer any shear forces. There is therefore a considerable difference between the mechanical action restraining the gas tank and the conventional philosophy that planes of symmetry act as planes of restraint.

C.11.10.4 In this work the mechanics of the elements representing the insulating support blocks has been modified from that described previously to accept and transmit horizontal shearing forces in the x and y directions all the time the load in the z direction remains compressive. At the current state of development of this element the element components in the three directions are added to the input data in the form of three springs. This could be modified so that the definition of one vertical support block would trigger the automatic inclusion of the two associated shear springs. The present method used in this work to define the direct and shear supports has the advantage that either a direct or shear support may be added without inclusion of the other. The program recognises the difference by considering the type number of the spring, a negative is automatically considered a shear support. When during the inversion stage of the program support blocks are being considered for deletion from the stiffness matrix an extra search is made if a block is to be deleted, to check if this direct support element is associated with a negatively **signed** spring if there is a positive result to the search then the springs thus found are also deleted from the stiffness matrix. Associated shear springs can be identified by the search routine by comparing the index of spring freedom numbers with the freedom numbers of each end of the direct spring which is about to be removed. During the allocation of global freedom numbers to the ends of springs, numbers must be given to shear springs directly before a direct support since allocation automatically proceeds from the the freedom in the local y direction to that in the local x direction and lastly to the local z direction which is the direct support. Therefore in the search, if the freedom being examined is one less than the direct support's number and has not been given a negative type then the search is abandoned because this direct spring must be acting alone. Conversely a shear spring not associated with a direct spring will never be removed from the stiffness matrix since there is no associated direct support to trigger the search. This provides a method of adding a stiffness term between two freedoms, should the need arise. This is in addition to the anchor type springs already developed whose effects also are not deleted from the stiffness matrix but which do not have both ends defined by freedom locations.

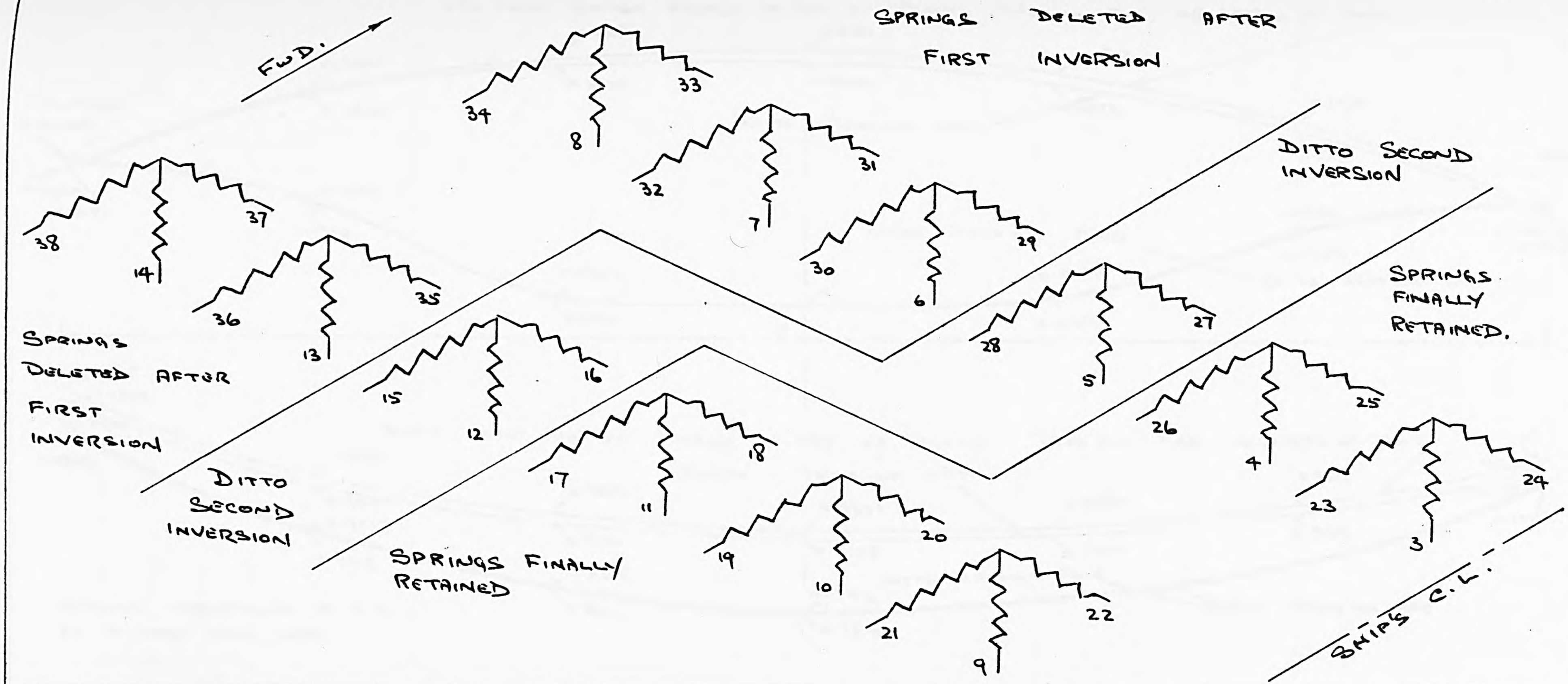
C.11.10.5 It is not sufficient to consider that a spring's contribution once deleted from the stiffness matrix may be ignored for the remainder of the inversion stage. Test 2 provides an example of a spring previously deleted because it was in tension being brought back into compression due to the subsequent flexing of the structure. In test 2 springs 1 and 3 were after the first inversion found to be in tension and deleted. This left the cantilevers still joined by spring 2 as well as spring 4 which is axiomatically the only spring eventually to remain in the system. After the second inversion it was found that the effect of spring 2 was to pull upwards the tip of the lower cantilever and similarly downwards the area near the root of the upper cantilever so that the first spring originally deleted is now in compression and its contribution to the stiffness matrix contribution replaced. To allow for this the inversion package, after each inversion stage scans all the spring ends to establish if any springs should be brought back. In the case of test 2 the next inversion which includes spring 1 produces a result having only spring 4 in compression. The presence of the springs transmitting shear loads whoes contributions depend upon whether their associated direct acting spring is still in contact requires the package, when searching for springs to replace, to identify the spring type and replace shear types only if its direct spring has gone from no contribution to apparent compression.

C.11.10.6 Test 2 was run with shear transmitting springs added at the thermal springs 1, 2 and 3. This necessitated the addition to the plate cantilevers of a stiffener to provide a degree of freedom normal to the plate. This modified problem gave as expected the same results as Test 2. The problem was also satisfactorily run with the stiffeners defined as the plate and the original plate cantilevers as the stiffeners.

C.11.11 Results.

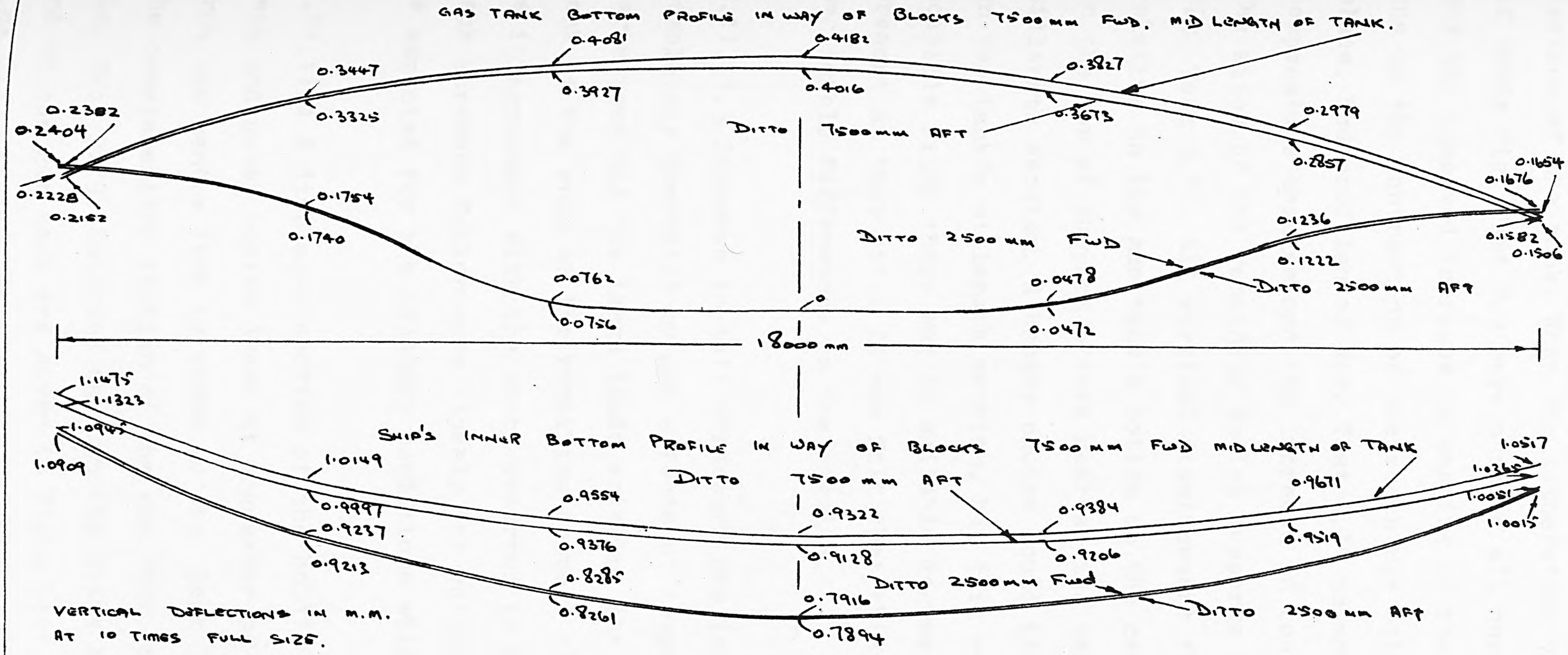
C.11.11.1 This problem was run for two conditions, one with no temperature change allowed anywhere in the structure and the other using the temperature differences given in C.11.8.1

C.11.11.2 In the case of no temperature changes, all 36 springs used for the support of the quarter structure remained in contact with both the tank and the inner bottom. When allowance was made for temperature variations 15 springs remained finally in contact with both the ship and the gas tank structure. 12 springs were deleted after the first inversion. For this particular set of data only three inversions were sufficient to establish the final displacements of the spring system. The order of the removal of springs is shown in Fig. Test 9.15. The profiles of the ship's inner bottom and gas tank with no temperature variation throughout the structure is shown in Fig. Test 9.16. It can be seen that near the midlength position of the tank where it is well supported by the ship's double bottom, the tank sags but hogs inway of the row of blocks furthest from the midlength



Numbering of insulating supports, indicating order in which supports were deleted and those which were finally retained.

Fig. Test 9.15



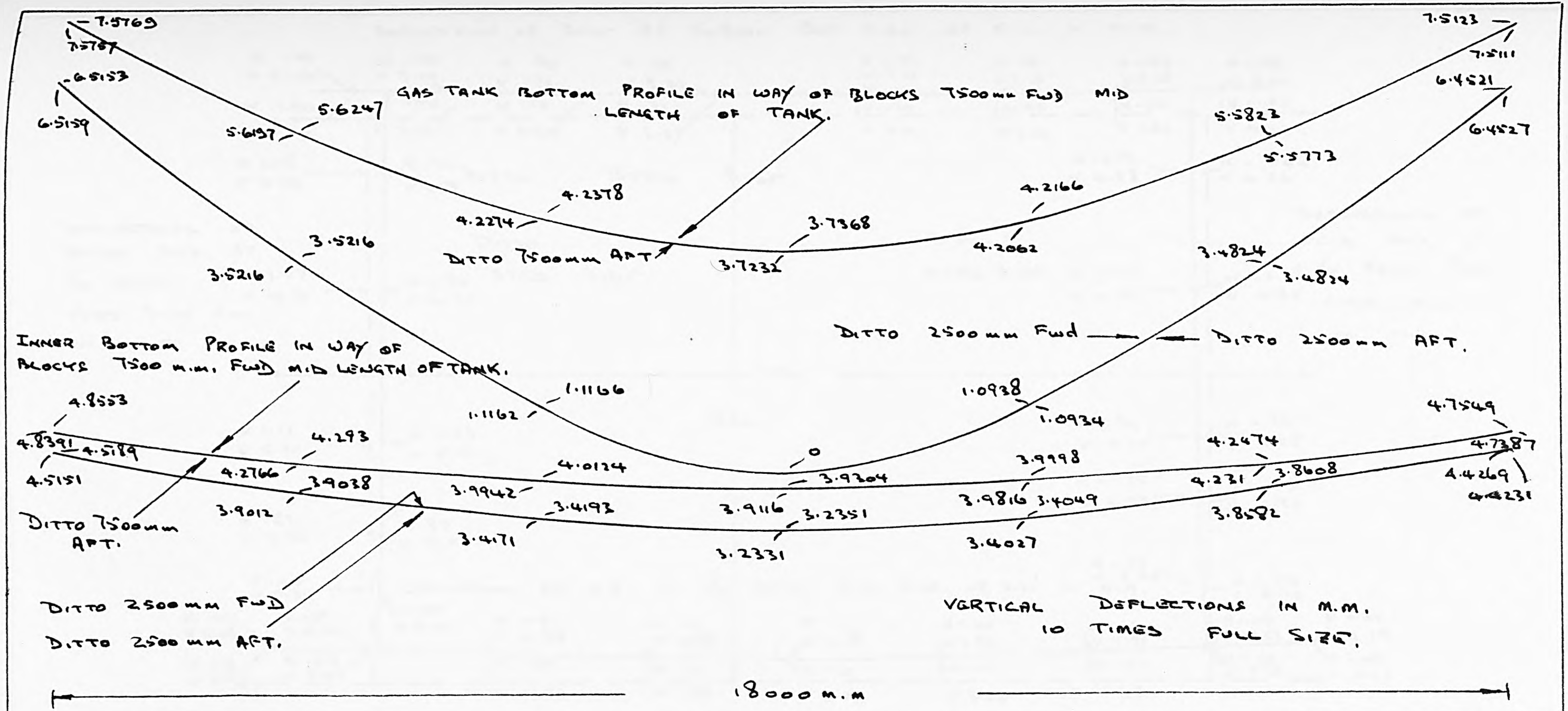
Inner bottom and Gas tank profiles in way of insulating support blocks.
No allowance made for temperature variations.

Fig. Test 9.16

section of the tank. When the temperature variations of Table Fig. Test 9.9 were applied all hogging ceased and the expected increase in sagging of the tank's bottom due to the contraction of the eccentric stiffening took place. The profiles of Fig. Test 9.16 as modified by the temperature changes and the increase in local loading at the sites of the remaining active supports is shown in Fig. Test 9.17. All vertical displacement figures are relative to the gas tank's bottom on the centre line in the row of support blocks nearest the tank's midlength section. A better choice would have been right on the tank's midlength section, but this was not possible since there was no suitable degree of freedom present at that site. It was felt that this would make negligible differences to the results.

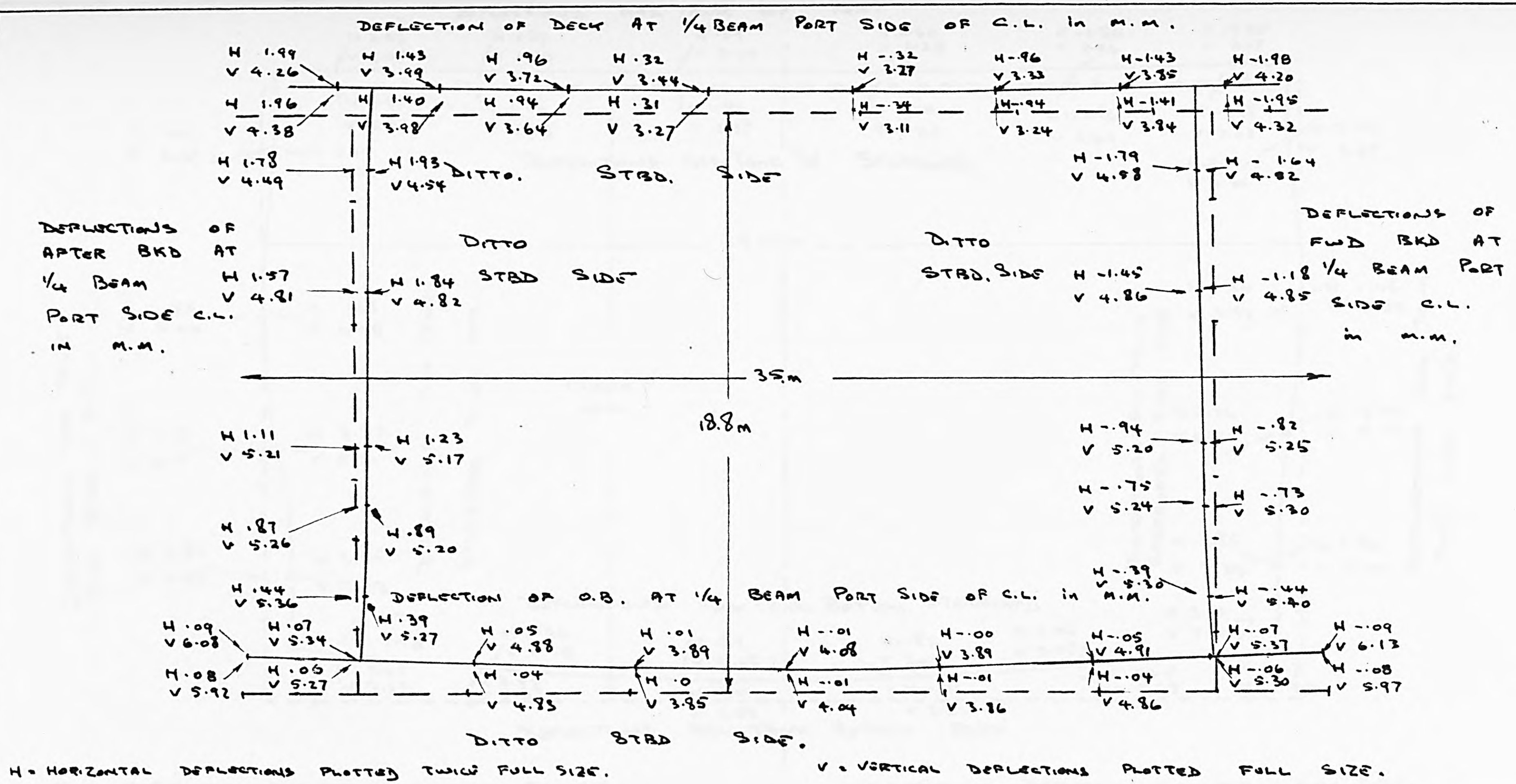
C.11.11.3 Stresses in this structure are in the main completely dominated by the effects of temperature variations and the large loads entering the hull and the tank at the ends of the remaining support blocks, which is in agreement with the state observed in practice. The deck stresses follow more closely the values that would be expected for the boundary conditions which were applied.

C.11.11.4 A displaced section of the hold's bulkheads, deck and outer bottom drawn at a quarter of the beam from the centre line is shown in Fig. Test 9.18 and the complementary section of the gas tank is given in Fig. Test 9.19. Sections through the sides of the ship and of the gas tank are given in Fig's Test 9.20 and Test 9.21 respectively. The effects of the low temperature



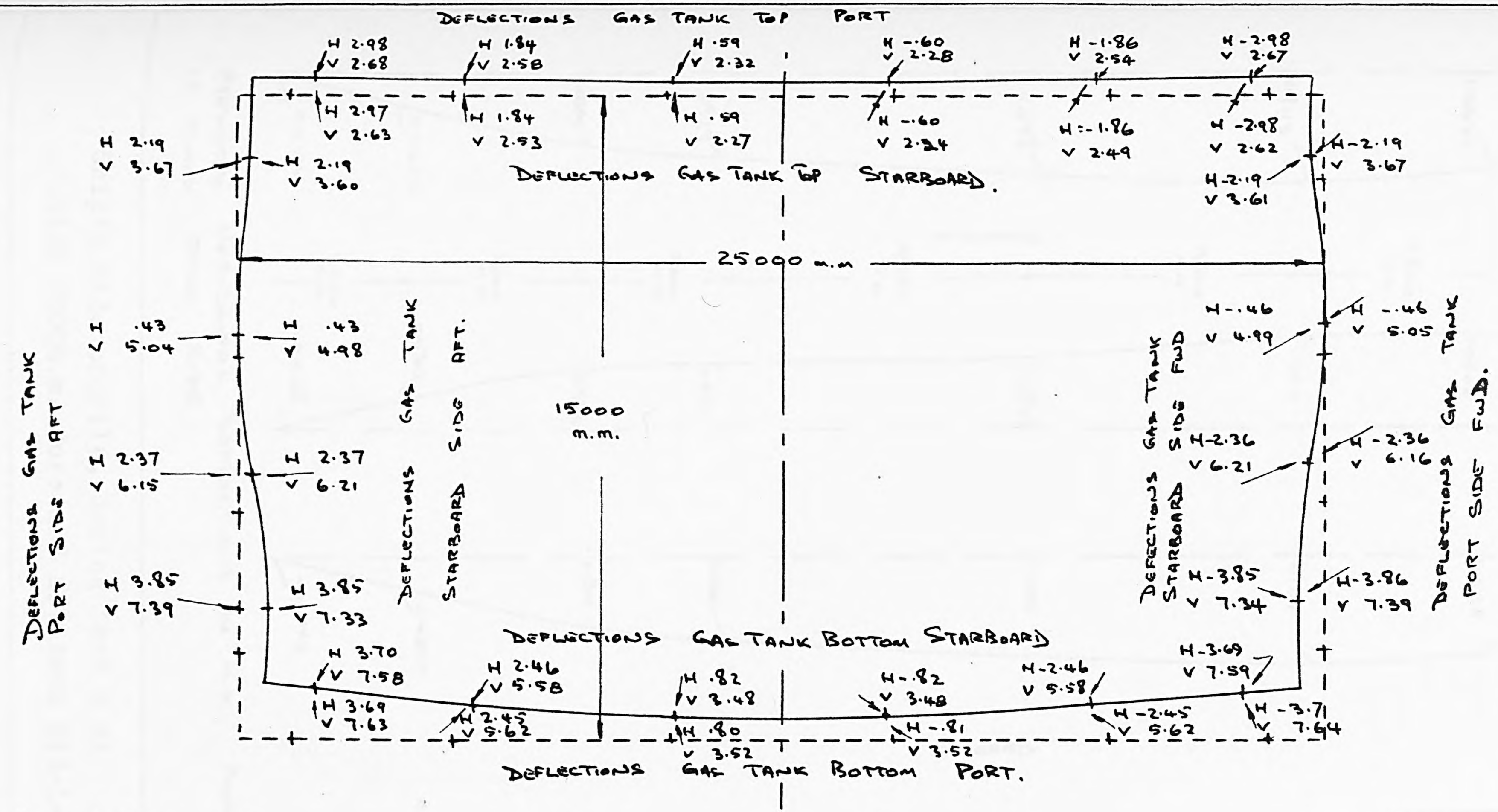
Inner bottom and Gas tank profiles in way of support blocks when structure is subjected to all the loads of Test 9. N.B. Fig. Test 9.16 is similarly loaded but does not have the temperature variations given in Table Fig. Test 9.9.

Fig. Test 9.17



Ship's deck, bulkheads and outer bottom deflections at $\frac{1}{4}$ beam, whilst the ship is subjected to all the loads of test 9

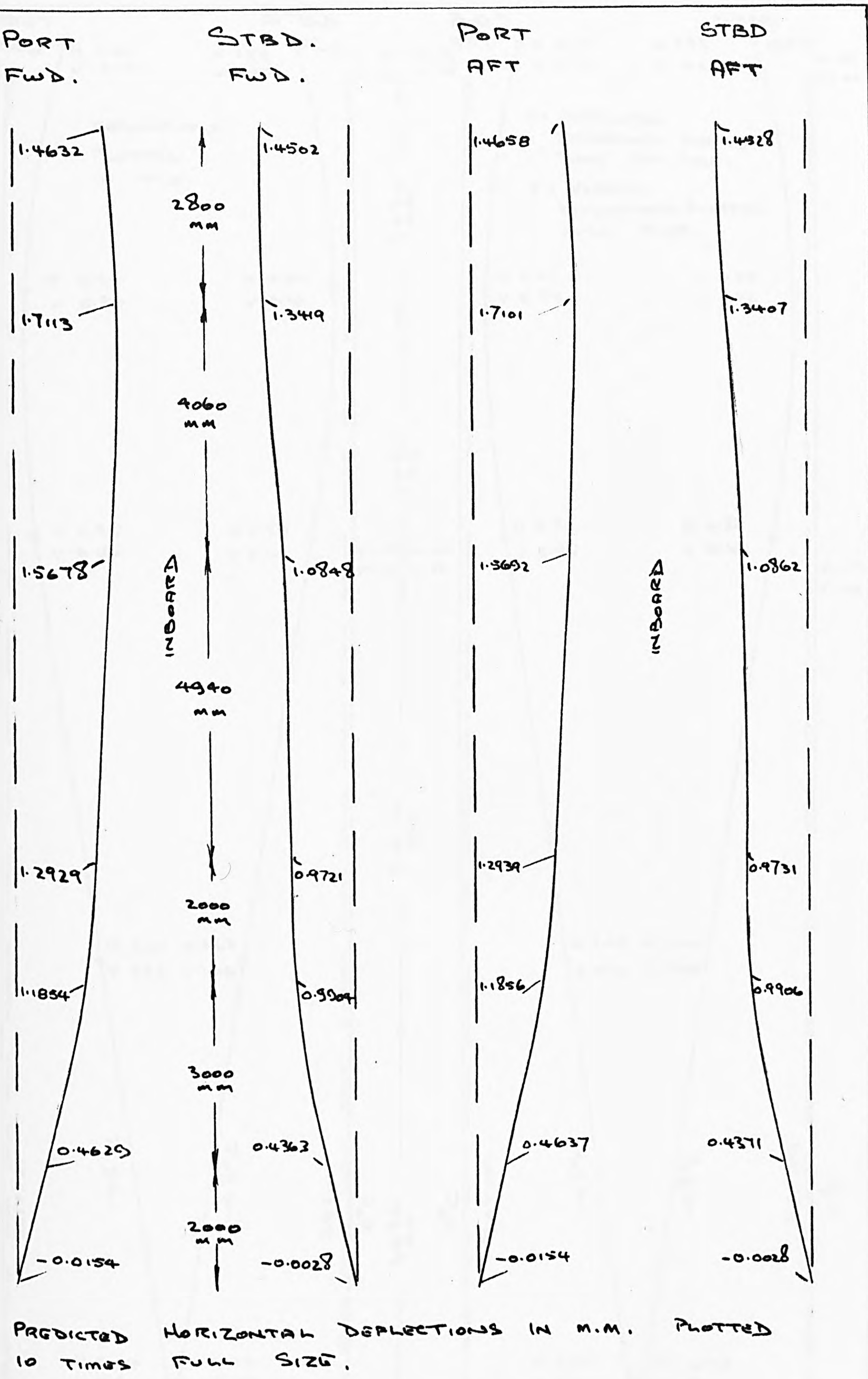
Fig. Test 9.18



H = HORIZONTAL DEFLECTION IN M.M. FULL SIZE V = VERTICAL DEFLECTION IN M.M. FULL SIZE.

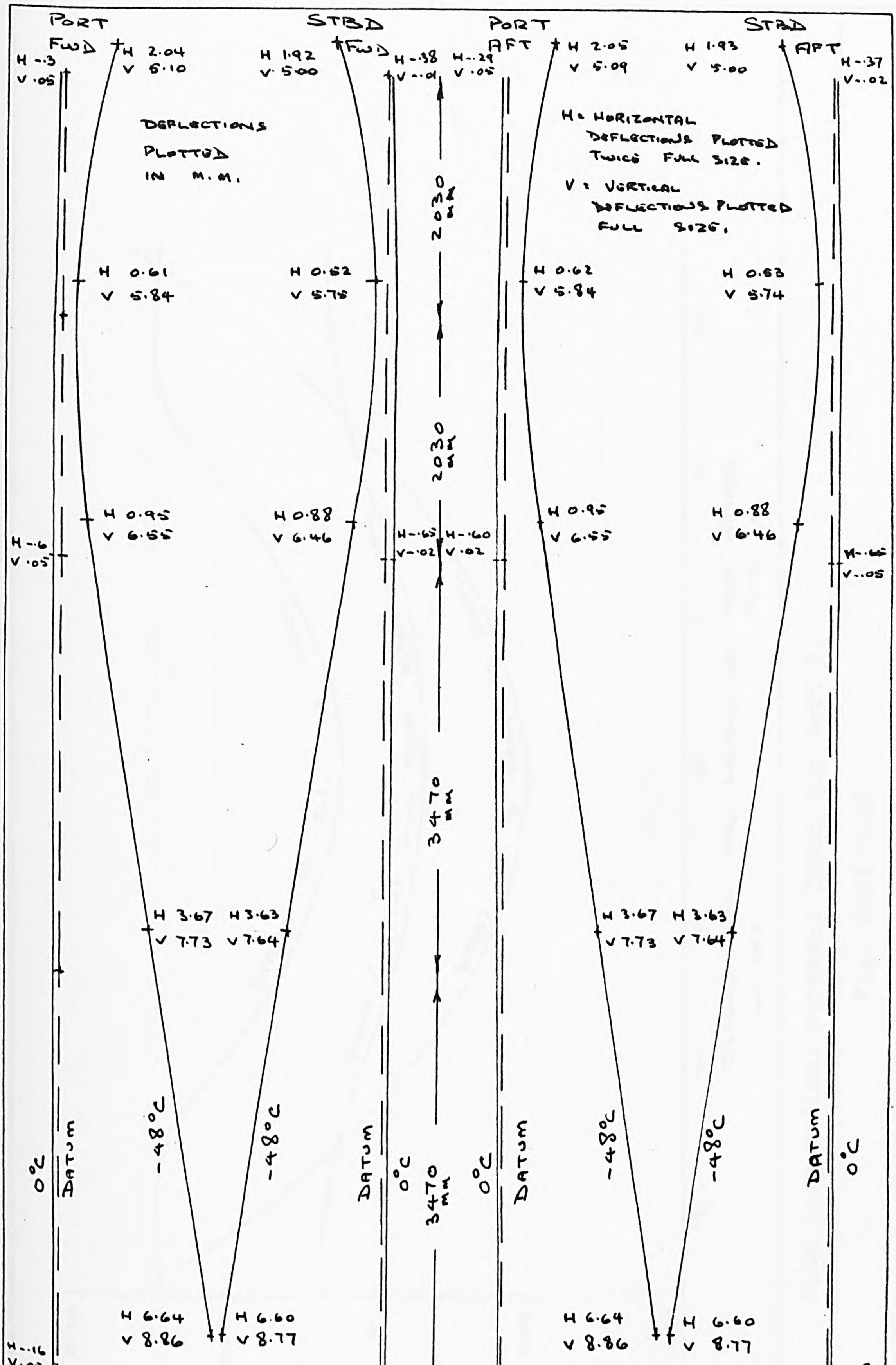
Predicted longitudinal section through gas tank at $\frac{1}{4}$ beam off C.L. during Test 9.

Fig. Test 9.19



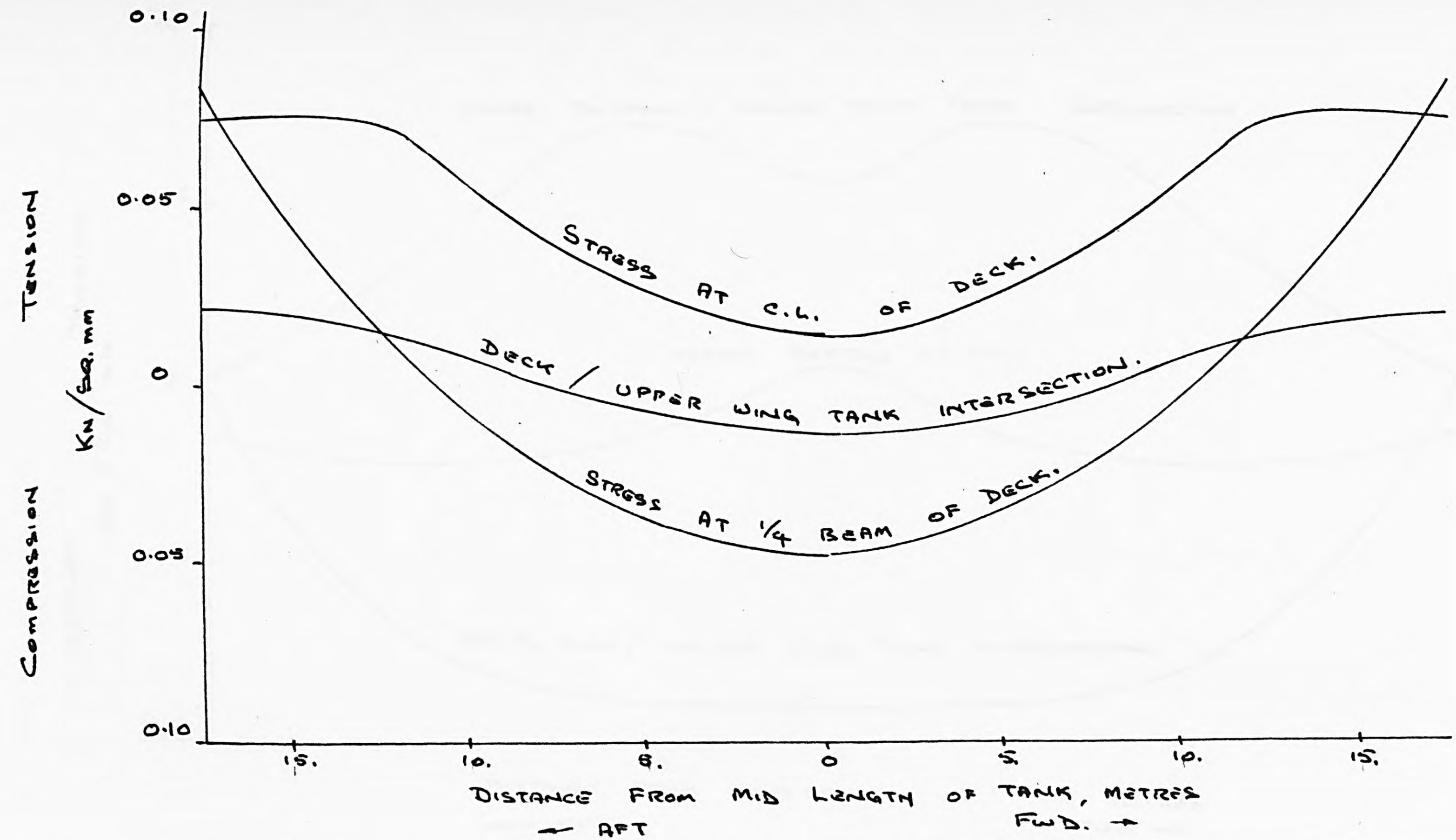
Ship's side profile, during test 9 at section 2500m.m. fore and aft tank mid-length.

Fig. Test 9.20.



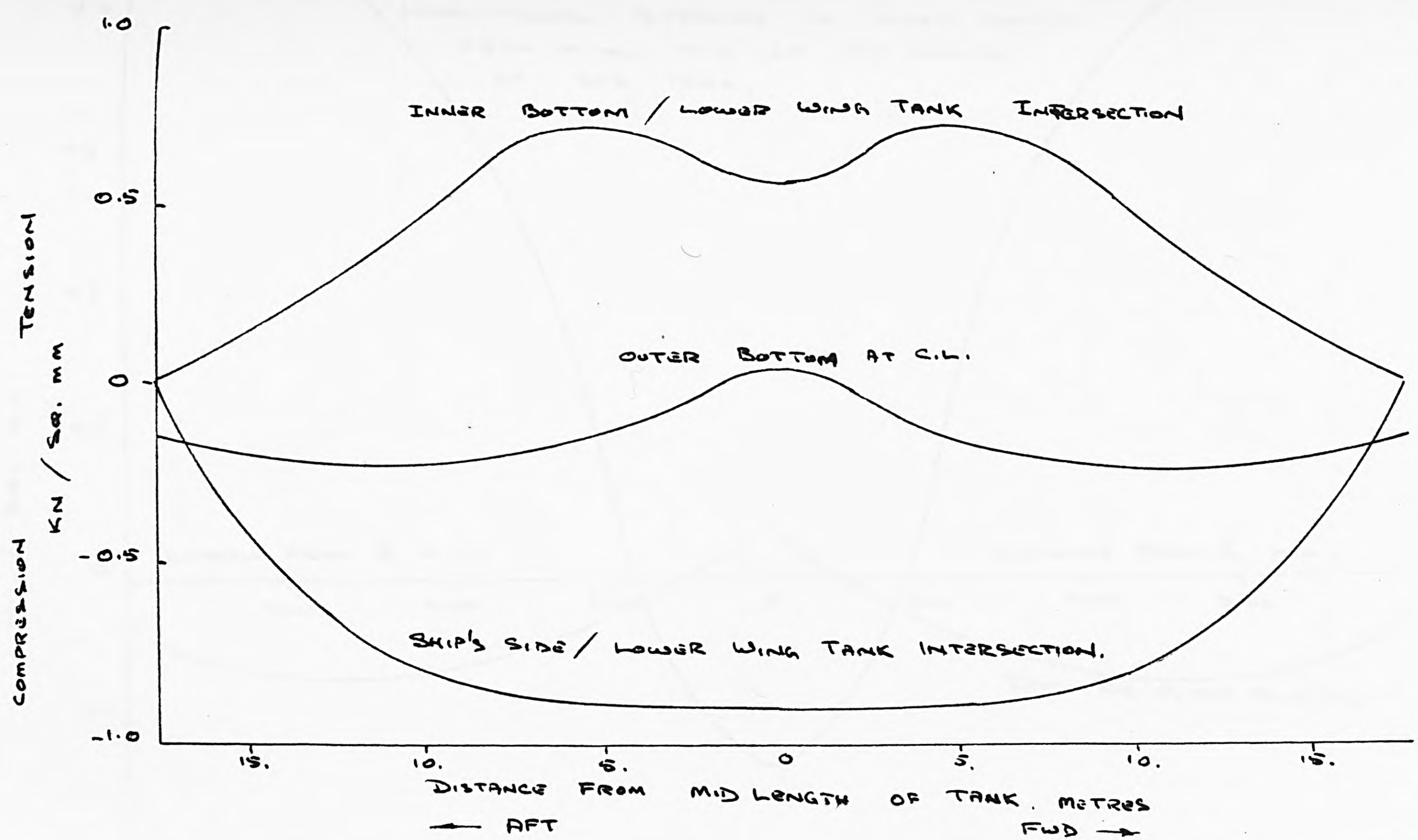
Profile of gas tank sides (knuckle to knuckle) with and without temperature variations of test 9.

Fig. Test 9.21



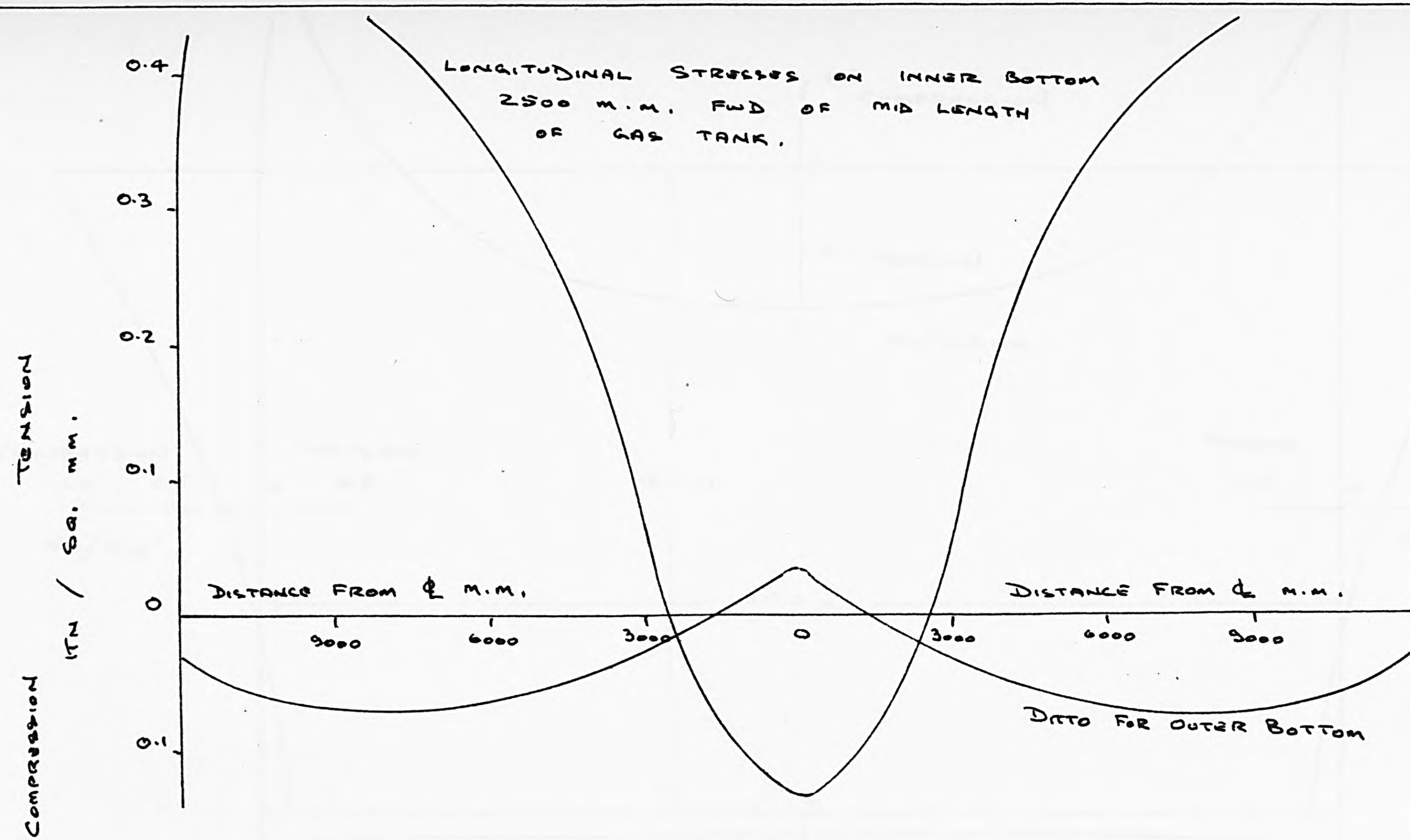
Deck longitudinal stresses found for test 9.

Fig. Test 9.22



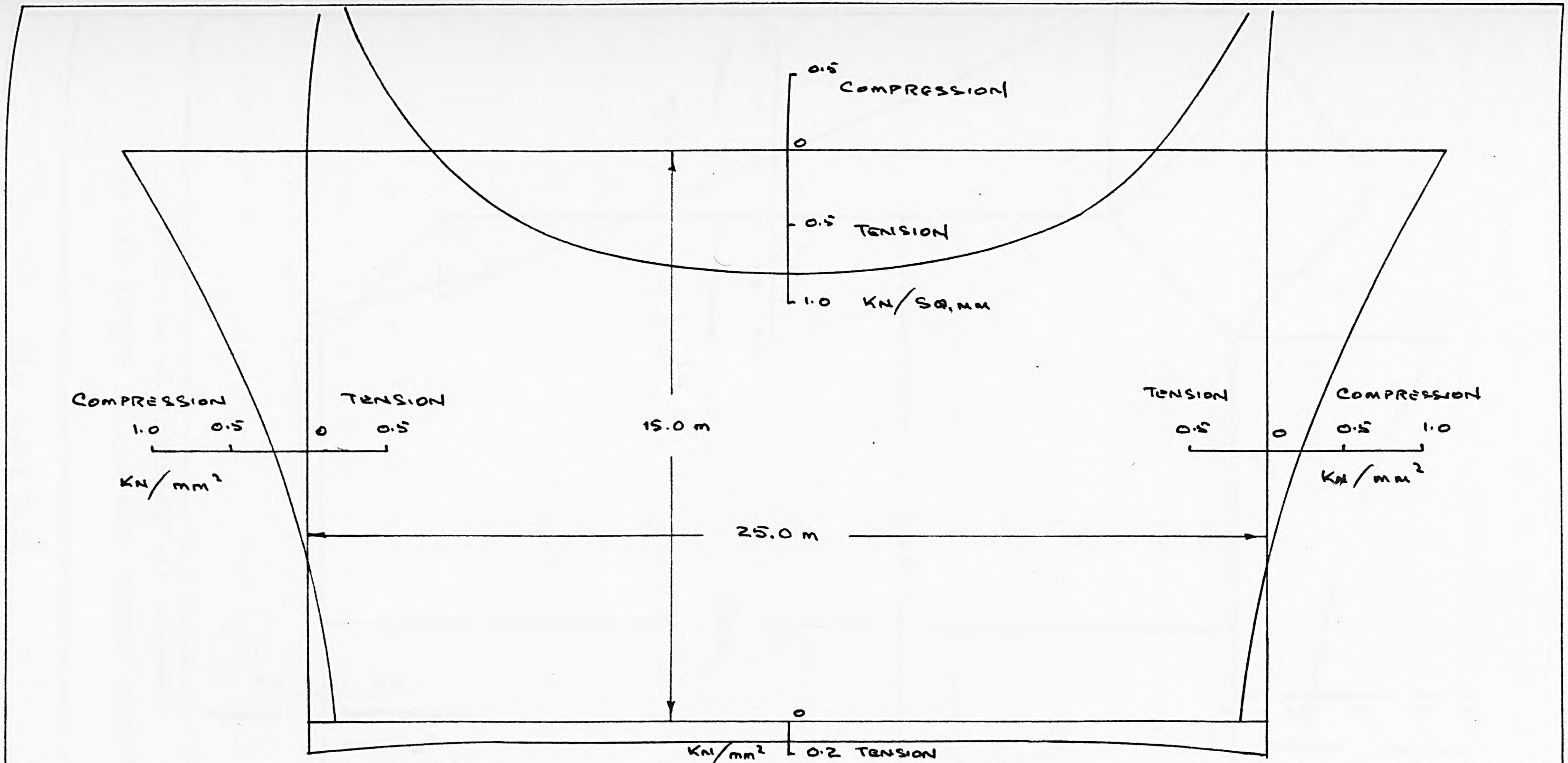
Outer bottom and lower wing tank longitudinal stresses for test 9.

Fig. Test 9.23



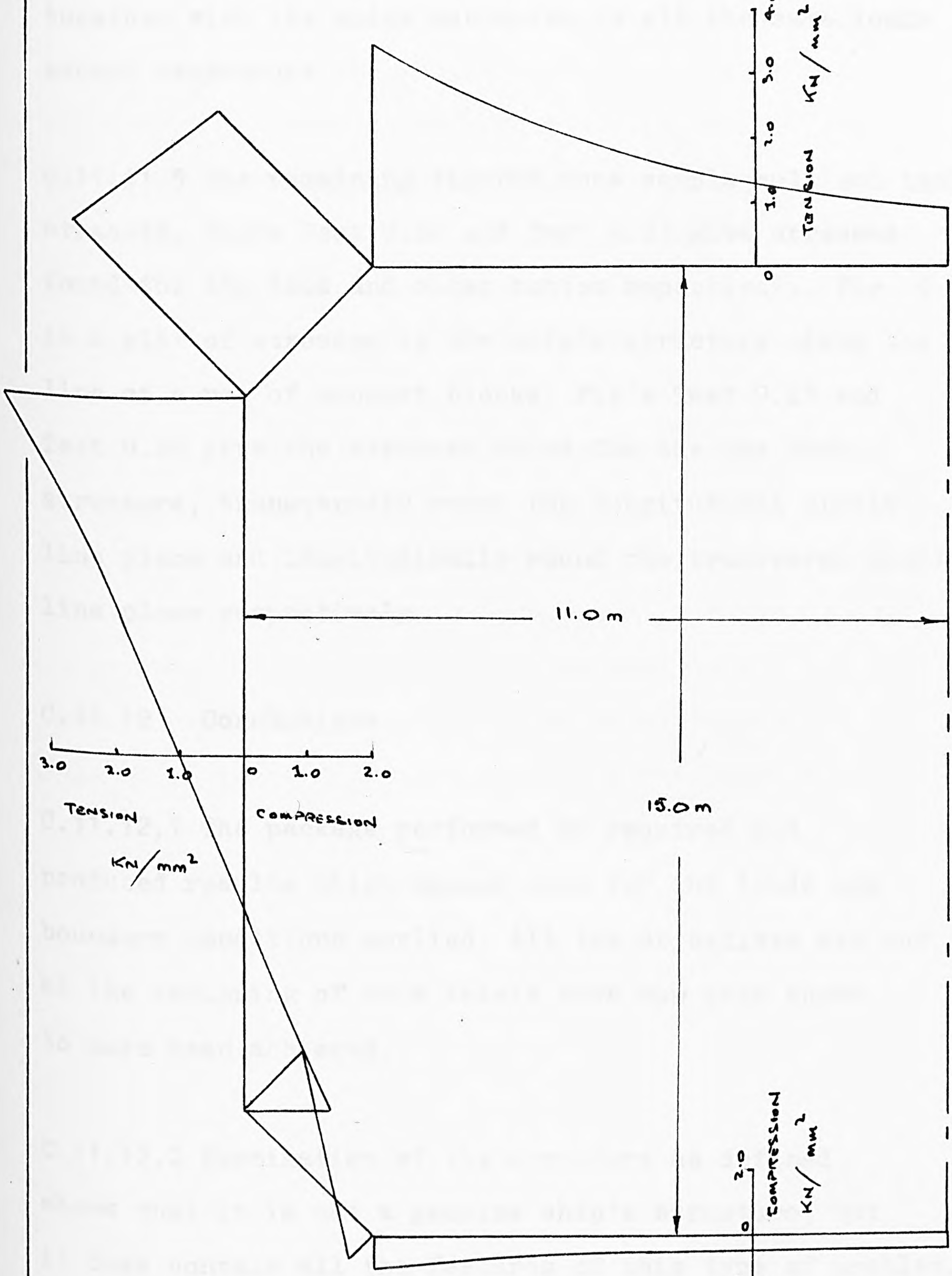
Inner and outer bottom longitudinal stresses in way of row of support blocks nearest midlength of gas tank.

Fig. Test 9.24



Transverse stresses round the C.L. plane of gas tank.
during test 9.

Fig. Test 9.25



Distibution of longitudinal stresses
at mid length of gas tank during test 9.

Fig. Test 9.26

are quite marked on the gas tank's sides and are shown together with the sides subjected to all the same loads except temperature.

C.11.11.5 The remaining figures show sample hull and tank stresses. Fig's Test 9.22 and Test 9.23 give stresses found for the deck and outer bottom repectively. Fig. 9.24 is a plot of stresses in the ship's structure along the line of a row of support blocks. Fig's Test 9.25 and Test 9.26 give the stresses found for the gas tank structure, transversely round the longitudinal middle line plane and longitudinally round the transverse middle line plane respectively.

C.11.12 Conclusions

C.11.12.1 The package performed as required and produced results which appear good for the loads and boundary conditions applied. All the objectives set out at the beginning of this thesis have now been shown to have been achieved.

C.11.12.2 Examination of the structure as defined shows that it is not a genuine ship's structure, but it does contain all the features of this type of problem. It was felt that it was sufficient to demonstrate the working of the package for checking a given structural design. The bowing of the ship's bulkheads symmetrically about the midlength of the hold, shown in the results indicates that the overlap into the adjacent spaces was not sufficient. It would be better to consider at least half of the next gas tank in the analysis. This would however put the problem outside the time limits allowed by the City University but still within the core space limits. A faster alternatve might be for the analysis of a single hold, to assume the holds on each side behave in a similar way and that the hold bulkheads may be restrained in the vertical plane.

C.11.12.3 There are no independent results available to quantify the accuracy of the results. Since other tests covered all the package's features and have shown excellent comparisons with independent results, there is no reason to believe that these results are not good for the twenty thousand degrees of freedom of the full hull and tank problem which has been solved.

Appendix D - Program listing (pp.489-615) has been removed for copyright reasons

References

1. Bass, R.L., Hokanson, J.C., Cox, P.A. A study to obtain verification of liquid natural gas tank loading criteria. Project SR - 218. U.S. Coastguard Headquarters, Washington D.C. 1976.

2. Unpublished Civil Service internal communication. London. 1976.

3. Report of the Sixth International Ship Structure Committee. Boston, U.S.A. 1976.

4. Report of the Fifth International Ship Structure Committee. Hamburg, Germany. 1976.

5. "The time-bomb on London's doorstep". New Scientist 5th November 1981.

6. Code for the construction and equipment of ships carrying liquefied gases in bulk. Inter-Governmental Maritime Consultative Organization. (Now the International Maritime Organisation). 1976.

7. Code for existing ships carrying

liquefied gases in bulk. Inter-Governmental Consultative Organization (Now the International Maritime Organization). 1976.

8. Carriage of liquefied gases in bulk, issue of certificates as to suitability of ships to carry such gases. Department of Trade, Merchant Shipping Notice No. M841, May 1978.

9. Report of "The Board of Investigation to Inquire into Design and Methods of Construction of Welded Merchant Vessels", U.S. Government Printing Office, Washington, 1947.

10. Sea transport of liquefied gases. Notes from course run by British Shipbuilders 1982.

11. Ffooks, R.C. A review of current LNG ship technology and an attempt to rationalise. Paper read at the LNG5 conference, Dusseldorf 1977.

12. Principal Regulations on Merchant Shipping. Department of Trade, Merchant Shipping Notice M.1039 September 1982.

13. Honnor, A.F., Andrews, D.J. — HMS Invincible, the first of a new genus of aircraft carrying ships. The Naval Architect. January 1982.

14. Senol Utku, M. — Systematic Substructuring. ASCE, Sixth Conference on Electronic Computation. Georgia Institute of Technology, Atlanta Georgia USA 1974.

15. Meyer, C. — Solution of Linear Equations - state of the art. Journal of the Structural Division, American Society of Civil Engineers, Vol. 99, No. ST7, July 1973.

16. Felippa, C.A. — Solution of Linear Equations with Skyline - Stored Symmetric matrix. Computers and Structures, Vol. 5, 1975.

17. Cheung, Y.K., Khatua, T.P. — A Finite Element solution program for large structures. International Journal for Numerical Methods in Engineering. Vol. 10, 1976.

18. Lyons, P. — Lusas - Finite Element Stress Analysis system. Users Manual. Finite Element Analysis Ltd., 25 Holborn Viaduct, London. 1980.

19. Bathe, K.J., Wilson, E.L., — Sap IV - A structural analysis program for static and dynamic response of

Peterson, F.E. linear systems. Users Manual. College of Engineering, University of California, Berkley, California, USA. Revised April 1974.

20. John, W. The strength of Iron Ships. Transactions of the Institute of Naval Architects, 1874.

21. Read, T.C., Jenkins, P. On the transverse strains in Iron Merchant Ships. Transactions of the Institute of Naval Architects, 1882.

22. Turner, M.J., Clough, R.W., Martin, H.C., Topp, L.J. Stiffness and Deflection analysis of complex structures. Journal of Aeronautical Sciences. Vol. 23 No. 9 1956.

23. Argyris, J.H. Energy theorems and structural analysis. Butterworth Scientific Publications. London 1960.

24. Clough, R.W. The Finite Element method in plane stress analysis. Proceedings of the second American Society of Civil Engineers Conference on Electronic Computation. Pittsburgh, Pen. Sept 1960.

25. Yuille, I.M., Wilson, L.B. Transverse strength of single hulled ships: elastic analysis by the displacement method. Trans. RINA, 1960.

26. Hughes. T.J.E. Some current trends in Finite Element research. Applied Mechanics Review 1980.

27. Fredriksson, B., Mackerle, J. Overview and evaluation of some versatile general purpose finite element computer programs. Paper 19 of "Finite Element Methods in the commercial environment". Ed. Robinson J. Pub. Robinson and Associates. 1978.

28. Viner, A.C. Milton, D. Ship structural analysis and interactive graphics. Shipping World and Shipbuilder. Aug. 1974.

29. Tocher, J.L., Herness, E.D. A critical view of Nastran. Numerical and Computer Methods in Structural Mechanics. Ed. Fenves, S.J. et al. Pub. Academic Press. 1973.

30. Zienkiewicz, O.C. The Finite Element Method. 3rd Ed. Pub. McGraw-Hill Inc. 1977.

31. Wilson, E.L., Taylor, R.L., Doherty, W.P., Ghabussi, T. Incompatible displacement models in numerical and computer methods. Structural Mechanics. Ed. Fenves, S.T. et al. Academic Press, 1973.

32. Przemieniecki, J.S. Theory of matrix structural analysis. Pub. McGraw-Hill Inc. 1968.

33. Olson, M.D., Bearden, T.W. — A simple flat triangular plate element revisited. International Journal of Numberical Methods in Engineering. Vol 14, No. 1. 1979.

34. Jan, H.Y., Liu, D., Shapiro, A.D. — Structural analysis of LNG vessels in a seaway. International conference on computer applications in automation of shipyard operation and ship design. Tokyo, Japan. Aug. 1973.

35. Hughes, T.J.R. — Recent developments in computer methods for structural analysis. Nuclear Engineering and Design, No 57. 1980.

36. Becker, H., Calao. — Thermoelastic model studies of cryogenic tanker structures. Ship Structures Committee report SSC-241. U.S. Coastguard Headquarters, Washington D.C. 1973.

37. Meriam, J.L., Lyman, P.T., Steidel, R.F., Brown, G.W. — Thermal stresses in SS Boulder Victory. Journal of Ship Researdh, Vol. 2, No. 2. Oct. 1958.

38. Irons, B.M. — A frontal solution program for finite element analysis. International Journal for numerical methods in Engineering, Vol. 2. 1970.

39. Sabir, A.B. — The nodal solution routine for the large number of linear simultaneous equations in the finite element analysis of plates and shells. Chap. 5 of Finite elements for thin shells and curved members. Pub. Wiley. 1976.

40. Melosh, R.J., et al. — Computer analysis of large structural systems. Journal of spacecraft and rockets. Vol. 6, No. 4, April 1969.

41. Hinton, E., Owen, D.R.J. — Finite element programming. Pub. Academic Press, 1977.

42. Klyuyev, V.K., Kokovkin-Scherbak, N.I. — On the minimization of the number of arithmetic operations for the solution of linear algebraic systems of equations. Technical report CS 24, Computer Science Dept., Stanford University, Stanford, California. 1965.

43. Melosh, R.J., Banford, R.M. — Efficient solution of load-deflection equations. Journal of the Structural Division. Proceedings of the American Society of Civil Engineers. April 1969.

44. Kark, S.K., Scholes, A. — Automatic constraining of linear dependencies. International Journal for numerical methods in engineering. Vol. 18. 1982.

45. Paulling, J.R. The analysis of complex ship structure by the finite element technique. Journal of Ship Research. Dec. 1964.

46. Mondkar, D.P., Powell, G.H. Towards optimal in-core equation solving. Computers and structures. Vol. 4. 1974.

47. Meyer, C. Special problems related to linear equation solvers. Journal of the Structural Division, Proceedings of the American Society of Civil Engineers. April 1975.

48. Fox, L., Mayers, D.F. Computer methods for scientists and engineers. Pub. Clarendon Press. 1968.

49. Krishnamoorthy, C.S., Selvanathan, J. Development of substructuring technique for finite element analysis of off-shore structures. Indian Institute of Technology, Technical Report No. 6. Jan. 1980.

50. Schrem, E. Implementation of finite element procedure. Numerical and Computer Methods in Structural Mechanics. Fenves, S.J. et al editors. Pub. Academic Press, Inc. 1973.

51. Finifter, D., A solution for the integration of

Wyniecki, P., DeCastel, J. finite element analysis in a ship design environment. Nuclear Engineering and Design, No. 70. 1982.

52. Swanson, J.A. Use of mini-computers for large scale structural analysis. Computers and Structures. Vol. 7. April 1977.

53. Rammant, J.P. Micro-computer, finite element analysis in practice. International Conference on Engineering Application of the finite element method. Hovik, Norway. May 1979.

54. Wilson, E.L. The uses of micro computers in structural analysis. Computers and Structures. Vol. 12, 1980.

55. Foster, E.P., Storaasli, O.O. Structural analysis on a mini computer. Proceedings of the First International Conference on Engineering Software. Southampton University. Sept. 1979.

56. L.R. expands software range. Lloyd's List. May 1983.

57. Bennett, C.M., Goodno, B.J. Structural analysis on micro computers. Proceedings of the Second Conference on Computing in Civil Engineering. Baltimore, USA. June 1980.

58. Clough, R.W. The finite element method after twenty five years - a personal view. Internation conference on Engineering Application of the finite element method. Hovik, Norway. May 1979.

59. Oden, Bathe. Commentary on computational mechanics. Applied Mechanics Review. Vol. 31, No. 8. Aug. 1978.

60. Bulkhead analysis. Unpublished report of a computer structural analysis. Department of Trade. 1980.

61. Tarpy, J.S. A technique for the finite element analysis of large complex structural systems. Ph. D thesis at Vanderbilt University, Tennessee. 1969.

62. Latreille, P.J. Calculation method and structural details analysis of an LNG/LPG tanker. Bulletin Technique du Bureau Veritas, April 1983.

NOTE.

Reference has been made in several places in this thesis to the Department of Trade. It should be noted that after the bulk of the work was written responsibility for the marine matters mentioned herein was transferred to the Department of Transport.