

Oasys



Oasys GSA

Theory

Oasys

YOUR IDEAS BROUGHT TO LIFE

13 Fitzroy Street
London
W1T 4BQ
Telephone: +44 (0) 20 7755 3302
Facsimile: +44 (0) 20 7755 3720

Central Square
Forth Street
Newcastle Upon Tyne
NE1 3PL
Telephone: +44 (0) 191 238 7559
Facsimile: +44 (0) 191 238 7555

e-mail: oasys@arup.com
Website: oasys-software.com

Oasys GSA

© Oasys 1985 – 2017

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Contents

Notation	8
Degrees of freedom	10
Active degrees of freedom	10
Front ordering	10
Active degrees of freedom	11
Degrees of Freedom with no Local Stiffness	12
Analysis Options	13
Static & Static P-delta Analysis	13
Static P-delta	13
Modal	14
Modal P-delta	14
Ritz Analysis	15
Buckling	16
Model Stability Analysis	17
Non-linear Static Analysis	18
Matrix Solver Options	18
Long Term Analysis	20
Dynamic Relaxation Solver	20
Applied Displacements and Lagrange Multipliers	29
Axes	30
Composite	31
Composite Slabs	31
Effective Elastic Properties	33
Condition Number	33
Constraint & Constraint Equations	35
Constraint Equation	35
Joint	35
Rigid Constraints	35
Tied interfaces	36
Direction Cosines	37
Dynamic Response Analysis	38
Harmonic Analysis	38

Periodic Load Analysis	40
Linear Time-history Analysis	40
Footfall Analysis	41
Element axes	46
Section elements	46
Non-vertical elements	46
Spring elements	47
Grounded spring and mass elements	47
Cable elements	48
Link elements	48
2D element axes	48
3D element axes	49
Updated element axes	50
Elements	51
Bar and Rod Elements	51
Cable, Tie and Strut Elements	52
Beam Elements	53
Spring Elements	56
Damper Elements	59
Mass Elements	60
2D Elements	60
Element stiffness	61
Element Mass	74
Element Stiffness	74
Releases	75
Offsets	76
Spacer Elements & Sliding Cables	76
Error Norms	85
Statics	85
Dynamics and buckling	85
Explicit Time Integration	85
2D Element Stresses and Forces	86
Stress in 2D Elements	86
Stress-strain Relationships	91

Force in 2D Elements	93
III Conditioning	95
Interpolation on a Triangular/Quad Facet	96
Loading	98
Beam Loads	98
Beam Thermal Loads	99
Beam Pre-stress Loads and Lack-of-fit	99
Loads on pin-ended beams	99
Projected Loads	100
2D Body Loads	100
2D Face Loads	101
2D Element Thermal Loads	101
2D In-plane Loads	102
Grid Loads	102
Lagrange Interpolation	107
Mass Distribution	108
Material Models	109
Isotropic material	109
Orthotropic material	109
Missing Mass & Residual Rigid Response	111
Gupta Method	112
Participation Factor and Effective Mass	113
Patterned Load Analysis	114
Raft Analysis	117
Iteration	117
Piles	118
Convergence	119
Rayleigh Damping	119
RC Slab	120
Introduction	120
Data Requirements	120
Other Symbols Used in Theory	122
RC Slab Sign Convention	123
RC Slab Analysis Procedure	124

Reduced stiffness & P-delta	139
Seismic Calculation	140
Response Spectrum Analysis	140
Equivalent Static and Accidental Torsion Load	144
Rotation at the end of a bar (beam)	147
Force	147
Moment	149
Thermal	151
Pre-stress	151
Beam distortion	152
Second Moments of Area & Bending	152
Shear Areas	154
Thin-walled Sections	154
Solid Sections	157
Storey Displacements	158
Sub-model Extraction	160
Static analysis tasks	160
Modal dynamic and Ritz analysis cases	160
Response spectrum analysis	161
Torce Lines	162
Torsion Constant	164
Saint Venant's Approximation	165
Rectangular Sections	165
Other Sections	166
Factored Values for Concrete	168
Transformations	168

Notation

Symbol	Represents
t	Time
T	Period
M	Mass matrix
K	Stiffness matrix
K_g	Geometric stiffness
C	Damping matrix
u	Displacement vector
a	Acceleration vector
φ	Mode shape
λ	Eigenvalue or buckling load factor
Λ	diagonal eigenvalue matrix
\tilde{m}	Modal mass
\tilde{k}	Modal stiffness
\tilde{k}_g	Modal geometric stiffness
f, \mathbf{f}	Force or force vector
f	Frequency
ω	Angular frequency
Γ	Participation factor
μ	Effective mass
a	Dynamic amplification
α, β	Rayleigh damping coefficients
ξ	Damping ratio

In general

scalar quantities are denoted by italics – e.g. mass m or mass M

vector quantities are denoted by lower case upright characters – e.g. displacements u

matrix quantities are denoted by upper case, upright characters – e.g. stiffness K

Degrees of freedom

Active degrees of freedom

Before the stiffness matrix is assembled it is necessary to decide which degrees of freedom need to be included in the solution.

The nodes can be categorised as follows:

- Inactive – the node does not exist.
- Non-structural – the node is not part of the structure (e.g. orientation node).
- Active – the node is part of the structure.

Likewise the degrees of freedom can be categorised as:

- Non-existent – this degree of freedom does not exist because the node is undefined.
- Inactive – this degree of freedom exists but is not used (considered like a restrained node).
- Restrained – the degree of freedom exists and is part of the structure but it is restrained and so it is not active in the stiffness matrix.
- Slave – the degree of freedom is constrained (through being in a rigid constraint, or by a repeat freedom) to move relative to a master degree of freedom and so it is not active in the stiffness matrix.
- Active – this degree of freedom is active in the stiffness matrix.

In setting up a list of degrees of freedom the following operations are carried out:

1. All the nodes are assumed to be inactive.
2. Look at elements attached to nodes to see which degrees of freedom are required.
3. Remove the degrees of freedom that are restrained by single point constraints or global constraints.
4. Remove the degrees of freedom that are slaves.
5. Remove degrees of freedom that have no local stiffness.
6. Number the degrees of freedom.

Front ordering

The way in which the degrees of freedom are ordered is important in keeping the size of the stiffness matrix to a minimum. The methods offered in GSA are

- Geometric – the degrees of freedom are ordered along a vector. By default this vector is along the longest dimension of the model but can be adjusted by the user.

- Reverse Cuthill-McKee – use the bandwidth minimizer of the same name,. This looks at the connectivity of the structure by using graphs and reorders the degrees of freedom so as to minimize the bandwidth.
- Cuthill-McKee – similar to the Reverse Cuthill-McKee method (but generally less optimal).
- Approximate minimum degree – this is used for sparse solvers to avoid excessive fill-in when factorizing the stiffness matrix

Active degrees of freedom

The degrees of freedom are made active based on the elements attached at the nodes. The degrees of freedom will depend on the element type: These are summarised in the table below:

Element	Active degrees of freedom per node
Bar Cable	3 translational
Rod	3 translational + 1 rotational
Beam	3 translational + 3 rotational
Translational spring	3 translational
Rotational spring	3 rotational
Mass	3 translational
Mass with inertia	3 translational + 3 rotational
2D plane stress 2D plane strain Axisymmetric	Bilinear formulation: 2 translational Allman-Cook formulation: 2 translational + 1 rotational
2D bending	Mindlin 1 translational + 2 rotational MITC 1 translational + 3 rotational
2D shell	Mindlin 3 translational + 2 rotational MITC 3 translational + 3 rotational
2D wall element	3 translational + 3 rotational
3D brick	3 translational

Degrees of Freedom with no Local Stiffness

It is possible to construct a model and find that there is no stiffness associated with particular degrees of freedom, either for translation or rotation. For example a model made up of shell elements in a general plane 6 degrees of freedom will be assigned per node, but there is only stiffness in 5 of these. There are a number of approaches to avoid this problem.

Geometry based automatic constraints

At each node in the structure the attached elements are identified. A pseudo stiffness matrix associated with rotations is set up with a value of one on the diagonal if the element is stiff in that direction or zero if there is no stiffness. All off-diagonal terms are set to zero. The pseudo stiffness is transformed into the nodal axis system (so the off-diagonal terms are, in general, no longer zero) and added to a nodal pseudo stiffness matrix.

Once this has been done for all the attached elements an eigenvalue analysis of the resulting pseudo stiffness is carried out to reveal the principal pseudo stiffnesses and their directions. If any of the principal pseudo stiffnesses are less than the pre-set "flatness tolerance" then those degrees of freedom are removed from the solution and an appropriate rotation to apply to the stiffness matrix at the node is stored.

Stiffness based automatic constraints

This is similar to the geometry based automatic constraints but instead of a value of one or zero assigned to degrees of freedom the actual stiffness matrix is used. The resulting stiffness matrix is the same as would result from restraining the whole model except from the rotations at the node of interest.

Again an eigenvalue analysis is carried out to reveal the principal stiffnesses and their directions. If any of the principal stiffnesses are less than the pre-set "stiffness tolerance" then those degrees of freedom are removed from the solution and an appropriate rotation to apply to the stiffness matrix at the node is stored.

Artificial stiffness in shells

An alternative and cruder approach is to make sure that there is some stiffness in all directions by applying an artificial stiffness in the directions that are not stiff. This is done by constructing the element stiffness matrix for shell elements and then replacing the zeros on the leading diagonal with a value of 1/1000th of the minimum non-zero stiffness on the diagonal.

Since this approach introduces an artificial stiffness term that has not physical basis it should be used with care.

Analysis Options

Static & Static P-delta Analysis

The static analysis is concerned with the solution of the linear system of equations for the displacements, \mathbf{u} , given the applied loads. The applied loads give the load or force vector, \mathbf{f} . The elements contribute stiffness, \mathbf{K} , so the system of equations is

$$\mathbf{K}\mathbf{u} = \mathbf{f}$$

When non-linear elements such as ties and struts are introduced the analysis is no-longer linear and become iterative. GSA uses initial stiffness method to avoid creation of global stiffness matrix at each iteration. The system of equation to be solved at each iteration

$$\begin{aligned}\Delta \mathbf{u}^i &= \mathbf{K}^{-1} \mathbf{r}^{i-1} \\ \mathbf{u}^i &= \mathbf{u}^{i-1} + \Delta \mathbf{u}^i\end{aligned}$$

Where \mathbf{r}^{i-1} is residual force from the previous iteration $\mathbf{r}^{i-1} = \mathbf{f} - \mathbf{f}_{\text{int}}^{i-1}$ and $\Delta \mathbf{u}^i$ is the change in displacement for current iteration. This method sometimes requires a large number of iterations to converge to the solution and this may offset the advantage of constant stiffness. To improve the convergence speed acceleration scheme is applied to the solution strategy

$$\begin{aligned}\mathbf{u}^i &= \mathbf{u}^{i-1} + \alpha \Delta \mathbf{u}^i \\ \alpha &= \frac{\Delta \mathbf{u}^{i-1}}{(\Delta \mathbf{u}^{i-1} - \Delta \mathbf{u}^i)}\end{aligned}$$

The acceleration scheme estimates the ratio of the original tangent stiffness to local secant stiffness using the successive change in the displacements. To avoid inaccurate estimate the ratio, the acceleration scheme is only applied at every alternative iterations.

Static P-delta

The static P-delta analysis is similar to the static except that a first pass is done to calculate the forces in the elements. From these forces the differential stiffness can be calculated. The stiffness of the structure can therefore be modified to take account of the loading and the displacements are then the solution of

$$(\mathbf{K} + \mathbf{K}_g)\mathbf{u} = \mathbf{f}$$

The options allow for

- A single load case to be used as the P-delta load case
- Each load case to be its own P-delta load case

In the first case the first pass of the analysis solves

$$\mathbf{K}\mathbf{u}_{PD} = \mathbf{f}_{PD} \rightarrow \mathbf{K}_g$$

for the P-delta case: then for all the load vectors

$$(\mathbf{K} + \mathbf{K}_g)\mathbf{u} = \mathbf{f}$$

is solved for all displacements

In the second case there is a one for correspondence between P-delta load case and analysis case so

$$\mathbf{K}\mathbf{u}_i = \mathbf{f}_i \rightarrow \mathbf{K}_{g,i}$$

then for each case

$$(\mathbf{K} + \mathbf{K}_{g,i})\mathbf{u} = \mathbf{f}$$

Modal

The modal dynamic analysis is concerned with the calculation of the natural frequencies and the mode shapes of the structure. As in the static analysis a stiffness matrix can be constructed, but in a modal dynamic analysis a mass matrix is also constructed. The free vibration of the model is then given by

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{K}\mathbf{u} = 0$$

The natural frequencies are then given when

$$|\mathbf{K} - \lambda\mathbf{M}| = 0$$

The eigenvalue problem is then

$$\mathbf{K}\boldsymbol{\varphi} - \lambda\mathbf{M}\boldsymbol{\varphi} = 0$$

Or across multiple eigenvalues

$$\mathbf{K}\boldsymbol{\varphi} - \Lambda\mathbf{M}\boldsymbol{\varphi} = 0$$

where $\{\lambda, \boldsymbol{\varphi}\}$ are the eigenpairs - eigenvalues (the diagonal terms are the square of the free vibration frequencies) and the eigenvectors (the columns are the mode shapes) respectively.

Modal P-delta

The modal P-delta is similar to the normal modal analysis but takes into account that loading on the structure will affect its natural frequencies and mode shapes. In the same way as a static P-delta analysis a first pass is carried out from which the differential stiffness can be calculated. This is used to modify the stiffness matrix so the eigenproblem is modified to

$$(\mathbf{K} + \mathbf{K}_g)\boldsymbol{\varphi} - \Lambda\mathbf{M}\boldsymbol{\varphi} = 0$$

Ritz Analysis

Often the use of modal analysis requires a large number of modes to be calculated in order to capture the dynamic characteristics of the structure. This is particularly the case when the horizontal and vertical stiffnesses of the structure are significantly different (while the mass is the same). One way to circumvent this problem is to use Ritz (or Rayleigh-Ritz) analysis which yield approximate eigenvalues. While these are approximate they have some useful characteristics.

The eigenvalues (natural frequencies) are upper bounds to the true eigenvalues

The mode shapes are linear combinations of the exact eigenvectors

The number of Ritz vectors required to capture the dynamic characteristics of the structure is usually significantly less than that required for a proper eigenvalue analysis.

Ritz analysis method

A set of trial vectors based initially on gravity loads in each of the x/y/z directions. The subsequent trial vectors are created from these with the condition that they are orthogonal to the previous vectors. The assumption is that we can get approximations to the eigenvectors by taking a linear combination of the trial vectors.

So for trial vectors

$$\mathbf{X}_m = [x_1 \ x_2 \ x_3 \ \cdots \ x_m]$$

Let

$$\varphi = \mathbf{X}_m \mathbf{s} = \sum_{i=1}^m x_i s_i$$

and if the approximation to the eigenvalue is λ , the residual associated with the approximating pair $\{\lambda, \varphi\}$ is given by

$$\mathbf{r} = \mathbf{K}\varphi - \lambda \mathbf{M}\varphi$$

The Rayleigh-Ritz method requires the residual vector be orthogonal to each of the trial vectors, so

$$\mathbf{X}_m^T \mathbf{r} = \mathbf{X}_m^T \mathbf{K}\varphi - \lambda \mathbf{X}_m^T \mathbf{M}\varphi = 0$$

Substituting for φ from above gives

$$\mathbf{X}_m^T \mathbf{K} \mathbf{X}_m \mathbf{s} - \lambda \mathbf{X}_m^T \mathbf{M} \mathbf{X}_m \mathbf{s} = 0$$

or

$$\mathbf{K}_m \mathbf{s} - \lambda \mathbf{M}_m \mathbf{s} = 0$$

with

$$\mathbf{K}_m = \mathbf{X}_m^T \mathbf{K} \mathbf{X}_m$$

$$\mathbf{M}_m = \mathbf{X}_m^T \mathbf{M} \mathbf{X}_m$$

This eigenproblem is then solved for the eigenpairs $\{\lambda, s\}$ and then the approximate eigenvectors are evaluated from

$$\varphi = \mathbf{X}_m s = \sum_{i=1}^m x_i s_i$$

Ritz trial vectors

The algorithm as applied in a single direction is as follows:

Create a load vector \mathbf{f} corresponding to a gravity load in the direction of interest

Solve for first vector

$$\mathbf{K} \mathbf{X}_1^* = \mathbf{f} \quad \text{solve for } \mathbf{X}_1^*$$

$$\mathbf{X}_1^T \mathbf{M} \mathbf{X}_1 = 1 \quad \text{normalize } \mathbf{M}$$

Solve for additional vectors

$$\mathbf{K} \mathbf{X}_i^* = \mathbf{M} \mathbf{X}_{i-1} \quad \text{solve for } \mathbf{X}_i^*$$

$$c_j = \mathbf{X}_j^T \mathbf{M} \mathbf{X}_i^* \quad \text{for } j=1, \dots, i-1$$

$$\mathbf{X}_i = \mathbf{X}_i^* - \sum_{j=1}^{i-1} c_j \mathbf{X}_j \quad \text{orthogonalize } \mathbf{M}$$

$$\mathbf{X}_m^T \mathbf{M} \mathbf{X}_m = 1 \quad \text{normalize } \mathbf{M}$$

Buckling

The problem in this case is to determine critical buckling loads (Eulerian buckling load) of the structure. The assumption is that the differential stiffness matrix is a linear function of applied load. The aim of the buckling analysis is to calculate the factor that can be applied to load before the structure buckles. At buckling the determinant of the sum of the elastic stiffness and the critical differential (or geometric) stiffness is zero.

$$|\mathbf{K} + \lambda \mathbf{K}_{g,crit}| = 0$$

Using the assumption of differential stiffness a linear function of loads gives

$$\mathbf{K}_{g,crit} = \lambda \mathbf{K}_g$$

so the equation is

$$|\mathbf{K} + \lambda\mathbf{K}_g| = 0$$

and the eigenvalue problem is then

$$\mathbf{K}\phi = -\lambda\mathbf{K}_g\phi$$

Model Stability Analysis

When a structural model is ill-conditioned (as reported by the condition number estimate) it could be a result modeling errors in the model. These errors could be of two types:

- Some elements may not be well connected or could be badly restrained, e.g. beam elements spinning about their axis.
- Some elements very stiff compared with all other elements in the model, e.g. a beam element of short length but a large section.

To detect such errors, model stability analysis, which is a qualitative analysis intended to reveal the causes of ill conditioning in models, can be useful. The analysis calculates the smallest and largest eigenvalues and corresponding eigenvectors of the stiffness matrix, i.e. it solves the problem

$$\mathbf{K}\mathbf{u} = \lambda\mathbf{u}$$

for eigenpairs $\{\lambda, \mathbf{u}\}$. For each mode that is requested, element virtual energies are calculated for each element in the model. These are defined as follows.

The virtual strain energy s_e for large eigenpairs where

$$s_e = \mathbf{u}_e^T \mathbf{K}_e \mathbf{u}_e$$

and virtual kinetic energy v_e for small eigenpairs, defined as

$$v_e = \mathbf{u}_e^T \mathbf{u}_e$$

The virtual energies can be plotted onto elements as contours. Typically, for an ill conditioned model, a handful of elements will have large relative values of virtual energies.

- Where the ill conditioning is caused from badly restrained elements, such elements will have large relative virtual kinetic energies.
- If the ill conditioning is from the presence of elements with disproportionately large stiffnesses, then these elements will have large virtual strain.

The analysis also reports, in increasing order, the eigenvalues computed. For the case of badly restrained elements, there is usually a gap in the smallest eigenvalues. The number of smallest eigenpairs to be examined is given by the number of eigenvalues between zero and the gap.

Non-linear Static Analysis

The non-linear static solver works using the dynamic relaxation method. This is an iterative method which simulates a process of damped vibration in small time increments (cycles). This is a specialisation of the explicit time-history solution method. Fictitious masses and inertias are computed for each free node.

At each cycle the forces and moments which elements exert on each node are summed for the current displacements. The linear and angular accelerations of each node are computed from its fictitious mass and inertia, damping is applied to the node's current linear and angular velocities and the node's shifts and rotations are calculated for the cycle.

This process is repeated until it is terminated by the user or the solution has converged (the out-of-balance forces and moments (residuals) at every free node are less than target values).

If the damping is too high or the fictitious masses and inertias of the nodes are too large, their shifts and rotations at each cycle will be very small and many cycles will be needed to achieve a result. If on the other hand the damping is too low or the masses and inertias are too small, the simulated damped vibration becomes unstable.

The two cases of an unstable structure and of unstable simulated damped vibration can be distinguished by inspecting the results. When the structure is unstable, the element forces change little from cycle to cycle and the shifts of the nodes at each cycle may be very large but do not vary significantly from cycle to cycle. If the simulated damped vibration is unstable, the forces and nodal displacements oscillate wildly between cycles and usually increase to enormous values. The third case of stable simulated damped vibration converging to a stable solution can be recognised because the residuals and the shifts of the nodes decrease overall from cycle to cycle.

It should be noted that very few structures are so unstable that they do not eventually converge to a solution. Generally secondary effects become operative with large deflections and allow the structure to reach some kind of equilibrium.

Matrix Solver Options

There are two main approaches to the solution of the system of equations – direct solutions and iterative solutions. The iterative solutions can be split into ones that involve the full system matrix and element by element (EBE) methods.

For the direct (matrix) solutions there a number of options available. The general equation to be solved in GSS is

$$\mathbf{f} = \mathbf{K} \mathbf{u}$$

In all cases the fact that the matrix \mathbf{K} is symmetric and relatively sparse is exploited in the solution.

Once the matrix is factorized the solution of the equations is a straightforward back substitution in two passes.

Sparse Parallel Direct Solver

The sparse parallel direct solver uses a similar storage scheme as the sparse direct solver but factorizes the matrix in parallel, utilising multiple cores in CPUs. This makes use of the 'Pardiso' solver from Intel Math Kernel Library. Pardiso uses METIS based reordering for reducing fill-in and employs Bunch-Kauffman based pivoting for a sparse LDLT factorization.

Sparse Direct Solver

The sparse direct solver is another option which exploits the sparsity of the structure matrices. The solution method is similar to the active column solver in that the solution method is direct although the actual methods used are somewhat different. The sparse direct solver makes use of the approximate minimum degree (AMD) algorithm to order the degrees of freedom. This method is useful in minimizing the amount of fill when factorizing the matrices. The actual factorizing uses a sparse LDLT algorithm.

These algorithms have been developed at the University of Florida CISE (<http://www.cise.ufl.edu/research/sparse/>).

Conjugate Gradient Solver

The conjugate gradient solver exploits the sparsity of the matrix to the full by keeping an index of the non-zero terms in the matrix. This means that the factorizing which produces fill-in is no longer an issue. Conjugate gradient solvers work with a preconditioner – so are known as preconditioned conjugate gradient (PCG) solvers to improve the condition number of the 'matrix' leading to better convergence.

The significant difference with the PCG is that it is iterative. In theory the solution will converge in no more iterations than the number of degrees of freedom in the solution, however rounding in the calculations means that this cannot be guaranteed. Moreover the aim is to get the solution to converge in a much smaller number of iterations so the preconditioner is used to give faster convergence. The line Jacobi preconditioner is recommended in GSA.

The conjugate gradient method and the use of preconditioners are described in many text books and the user is directed to the bibliography for further information.

Active Column Solver

The active column solver only makes partial use of the sparsity of the matrix. The stiffness matrix is stored using skyline storage – only the above diagonal elements and that exist below the 'skyline' are stored. Some or many of the entries within the skyline may be non-zero however in the factorization there is fill-in which leads to these entries becoming non-zero. This type of storage works well with an active column solver where the factored matrix

$$\mathbf{K} = \mathbf{LDL}^T$$

where the (transpose of the) lower triangle \mathbf{L}^T and the diagonal \mathbf{D} can be stored within the same skyline profile.

Long Term Analysis

Long term analysis is not a different type of analysis as such. Instead it is an analysis where creep is taken into account for concrete materials. A creep coefficient is specified and this is used in the analysis to give effective E and G values for the concrete materials.

$$E_{eff} = \frac{E}{1 + \phi}$$
$$G_{eff} = \frac{G}{1 + \phi}$$

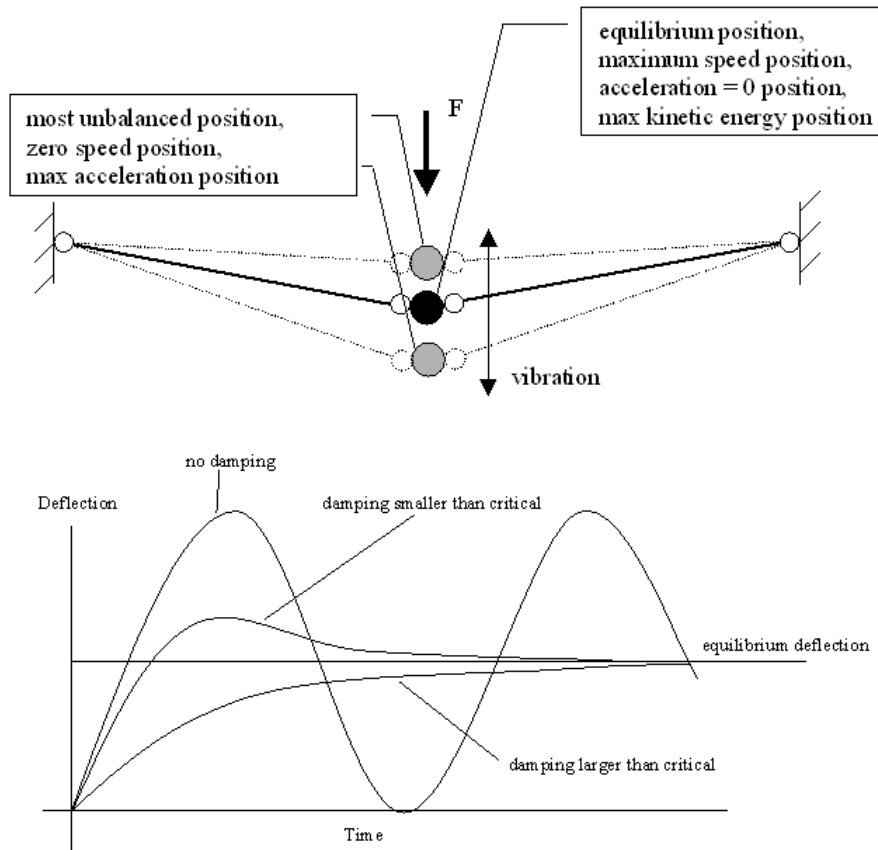
Dynamic Relaxation Solver

Dynamic relaxation is an analysis method for non-linear statically loaded structures using a direct integration dynamic analysis technique. In dynamic relaxation analysis it is assumed that the loads are acting on the structure suddenly, therefore the structure is excited to vibrate around the equilibrium position and eventually come to rest on the equilibrium position. In order to simulate the vibration, mass and inertia are needed for each of the free nodes. In dynamic relaxation analysis, artificial mass and inertia are used which are constructed according to the nodal translational stiffness and rotational stiffness. If there is no damping applied to the structure, the oscillation of the structure will go forever, therefore, damping is required to allow the vibration to come to rest at equilibrium position. There are two types of damping: viscous damping and kinetic damping. Kinetic damping is an artificial damping which will reposition the nodes according to the change of system kinetic energy.

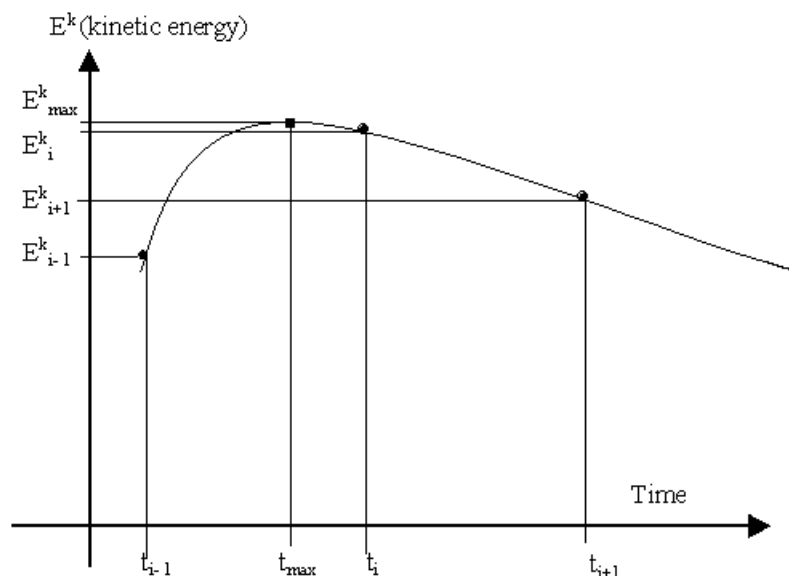
Damping

There are two types of viscous damping, one is viscous damping and one is artificial viscous damping. Viscous damping will apply the specified (or automatically selected) percentage of the critical damping to the system. Artificial viscous damping will artificially reduce the velocity at each cycle by the specified (or automatically selected) percentage of velocity in previous cycle. Once artificial viscous damping is used, kinetic damping will be disabled automatically. By applying one or both of these artificial damping methods, the vibration will gradually come to rest at the equilibrium position and this will be the solution given by dynamic-relaxation analysis.

The structure below shows the effect of viscous damping on the dynamic relaxation analysis process. The oscillation of the structure eventually comes to rest at the static equilibrium position if viscous damping is applied. The problem with viscous damping is that it is not an easy task to estimate the critical damping of the structure.



Kinetic damping is unrelated to conventional concepts of damping used in structural dynamic analysis. It is an artificial control to reduce the magnitude of the vibration in order to make it come to rest. It is based on the behaviour of structures with only one degree of freedom or the vibration of a multiple degree of freedom structure in a single mode. For these cases it is known that the structure's kinetic energy reaches a maximum at the static equilibrium position.



The structure's kinetic energy is monitored in the analysis at each time increment. The Kinetic energy increases as the nodes approach equilibrium position and starts to decrease once the

nodes are away from equilibrium position. Once the kinetic energy starts to decrease, an estimate of the equilibrium position of the nodes can be interpolated from the previous nodal positions and kinetic energies.

At this point the kinetic damping process is applied. The vibration is stopped and the nodes repositioned to correspond to the maximum kinetic energy. Assuming the relationship between structural kinetic energy and time is a parabola, then the moment at which the kinetic energy peaked can be calculated. Based on the previous nodal displacements and rotations, the equilibrium positions of the nodes can be estimated. After shifting the nodes to their optimum positions, the analysis will restart again by resetting the time, speed and acceleration to be zero.

Since it is unlikely that a multiple degree of freedom structure will vibrate in a single mode, it is impossible to find the equilibrium position just by reaching the maximum kinetic energy of the structure once or twice. Nevertheless, previous experience has shown that the use of kinetic damping is very efficient in searching for the equilibrium position in dynamic relaxation analysis.

Solution Process

The following steps are used in a dynamic relaxation analysis.

1. Compute equivalent nodal forces and moments. In this process, member loads are converted into nodal force or moments. These are the forces that initiate vibration.
2. Construct dummy mass and dummy inertia for the unrestrained (active) nodes according to the translational and rotational stiffness of the members at the nodes
3. Compute the acceleration, speed and displacement for each node at each cycle.
4. Compute a new nodal position and rotation for each node at each cycle; update the nodal stiffness and member force acting on the nodes.
5. Check the force and moment residuals at each node at the current position.
6. If no residual exceeds the limit, the analysis is considered to have converged and the final position is considered as the equilibrium position of the structure.
7. If any residual is not satisfied, the analysis is continued to the next step.
8. Compute the total kinetic energy of the structure. If the kinetic energy at a cycle overshoots the maximum, it is considered that the equilibrium position has been passed. Therefore, all nodes will be re-positioned so that they are closer to the equilibrium position. Reset the speed and acceleration to be zero and let the structure start to vibrate again from the new position.
9. After analysis has been converged, the element forces, moments and stresses are calculated according to the final equilibrium position of the nodes.

Fictitious masses and inertia

To speed up and simplify dynamic relaxation analysis, fictitious (dummy) masses and inertia rather than real masses and inertia are used in dynamic relaxation analysis. The fictitious masses and inertia are generated automatically in the solver. However, fictitious masses and inertia can be adjusted pre and during analysis by applying dummy mass and inertia factors and/or dummy mass and inertia power.

The fictitious masses and inertias calculated by the program are designed to be small enough for convergence to be reasonably fast but large enough to prevent nodes shifting too much in one cycle, which causes the solution method to become unstable. To this end, it is logical to take the fictitious masses and inertia proportional to the nodal translational stiffness and rotational stiffness respectively. From previous experience, it is found that the best estimate of the fictitious masses and inertia are two times the nodal translational stiffness and rotational stiffness respectively and they are calculated as follows

- Fictitious mass of a node = 2 × sum of translational stiffness of the elements connected to the node
- Fictitious inertia of a node = 2 × sum of rotational stiffness of the elements connected to the node

Control parameters

The iterative dynamic relaxation process continues until convergence criteria (unbalanced nodal force and moment) are met. If this does not happen, the iteration will continue for a maximum number of cycles, or analysis time in minutes.

It is almost impossible to achieve 100% accurate results in non-linear analysis, so an acceptable residual (tolerance) force and moment should be specified. The residual may be absolute or relative.

If relative residual is selected, the actual force residual and moment residual at each node are calculated from

$$\Delta f = \rho_f \frac{\sum f}{n}$$

$$\Delta m = \rho_m \frac{\sum m}{n}$$

where

$\Delta f, \Delta m$ are force residual and moment residual respectively

ρ_f, ρ_m are relative force residual and relative moment residual respectively

$\sum f$ is the sum of the total imposed loads including both nodal and member loads

n is the number of nodes in the structure

If there is no imposed load, e.g. a structure subjected only to support settlement, the force residual and moment residual are calculated from

$$\Delta f = 0.01 \times \rho_f \frac{\sum K_f}{n}$$

$$\Delta m = 0.01 \times \rho_m \frac{\sum K_m}{n}$$

where

$\sum K_f, \sum K_m$ are the sum of nodal translational stiffness and rotational stiffness of all the nodes in the structure.

If an absolute residual is selected, the specified force residual and moment residual will be used in the analysis.

Beam – Axial force

The axial force f_x of a beam is first calculated as

$$f_x = f_p + EA\varepsilon$$

where f_p is the pre-stress force.

If this force is greater than the yield capacity in tension it is set to the yield capacity in tension; if it is less than the yield capacity in compression it is set to the yield capacity in compression. The yield capacities are

$$f_{y,tens} = A\sigma_{y,tens}$$

$$f_{y,comp} = A\sigma_{y,comp}$$

where $\sigma_{y,tens}$ is the tensile yield stress and $\sigma_{y,comp}$ is the compressive yield stress

The strain is calculated as

$$\varepsilon = \frac{(\text{distance between nodes}) - (\text{unstressed length})}{(\text{unstressed length})}$$

The unstressed length is the initial distance between the end nodes (or the ‘initial length’ as specified by the user) modified for temperature.

Beam – Shear force and torsion

The shear modulus of a beam is assumed to be

$$G = \frac{E}{2(1+\nu)}$$

The shear strain caused by a shear force is considered to be uniform over the whole beam for planes normal to a principal axis. The shear strain between a principal axis and the local beam x axis is taken as

$$\varepsilon_{xy} = \frac{\sigma_{xy}}{G}, \quad \varepsilon_{xz} = \frac{\sigma_{xz}}{G}$$

and the effective shear stress is taken as

$$\sigma_{xy} = \frac{V}{k_y A} \quad \sigma_{xz} = \frac{V}{k_z A}$$

Where k_y, k_z are the shear factor along the principal axis closest to the local beam y/z axis.

The angle by which a beam is twisted about its local x axis is simply considered to be

$$\frac{M_x l}{GJ}$$

Beam – Axial force – flexural stiffness interaction

If slenderness effects are to be considered the bending stiffness of a beam is modified according to the axial load by using Livesey's 'stability functions'¹.

For a continuous/continuous beam within the elastic range the bending moment at end 1 is taken as

$$M_1 = \frac{EI(S \times \theta_1 + SC \times \theta_2)}{l} \quad \text{equation A}$$

S and SC are derived from series

$$S = \frac{\left(\frac{2}{3!} - \frac{4}{5!}k + \frac{6}{7!}k^2 - \frac{8}{9!}k^3 + \dots \right)}{\left(\frac{2}{4!} - \frac{4}{6!}k + \frac{6}{8!}k^2 - \frac{8}{10!}k^3 + \dots \right)}$$

$$SC = \frac{\left(\frac{1}{3!} - \frac{k}{5!} + \frac{k^2}{7!} - \frac{k^3}{9!} + \dots \right)}{\left(\frac{2}{4!} - \frac{4}{6!}k + \frac{6}{8!}k^2 - \frac{8}{10!}k^3 + \dots \right)}$$

where

¹ M.R. Horne & W. Merchant "The Stability of Frames" Pergamon 1965

$$k = \frac{f_x \times l^2}{EI}$$

and compression is positive.

For a continuous/pinned beam within the elastic range the bending moment at end 1 is taken as

$$M_1 = \frac{EI(S'' \times \theta_1)}{l} + C \times M_2 \quad \text{equation B}$$

S'' and C are derived from series

$$S'' = \frac{\left(1 - \frac{k}{3!} + \frac{k^2}{5!} - \frac{k^3}{7!} + \dots\right)}{\left(\frac{2}{3!} - \frac{4}{5!}k + \frac{6}{7!}k^2 - \frac{8}{9!}k^3 + \dots\right)}$$

$$C = \frac{\left(\frac{1}{3!} - \frac{k}{5!} + \frac{k^2}{7!} - \frac{k^3}{9!} + \dots\right)}{\left(\frac{2}{3!} - \frac{4}{5!}k + \frac{6}{7!}k^2 - \frac{8}{9!}k^3 + \dots\right)}$$

where M_2 is the moment at end 2 and compression is positive.

These series are used to pre-calculate S , SC , S'' and C for ten values of K . During calculation cycles values of S , SC , S'' and C are interpolated for the current value of K .

Beam – Yielding

For an explicitly defined section the bending moments about the principal axes are limited to the following value

$$0.45 \times (\sigma_{y, tens} - \sigma_{y, comp}) \times \sqrt{A \times I_p}$$

For equal yield stresses this is a good approximation to the plastic bending moment capacity.

The axial force is computed as above.

The calculations for the axial force and for the bending moments about the principal axes are all performed independently. Beams are assumed to behave elastically up to the limiting force or bending moment. Thus plastic behaviour is only modelled with any degree of realism for cases where either

- only axial forces are significant or
- only bending about one principal axis is significant, the tensile and compressive yield stresses are similar and the transition between first yield and full plasticity can be ignored.

If the bending moment at one end of a beam has been limited to the plastic moment capacity, the bending moment at the other end is obtained by using equation B above

$$M = \frac{EI \times S'' \times \theta}{l} + C \times M_{plas}$$

This bending moment is in turn limited to the plastic moment capacity.

For a beam with a standard shape section and a specified yield stress, the program calculates the tensile and compressive yield forces of the section, which are taken to be

$$f_{y,tens} = A\sigma_{y,tens}$$

$$f_{y,comp} = A\sigma_{y,comp}$$

The program then constructs a 'look up' table for each shape before the commencement of calculation cycles.

The 'look up' table contains values of

- bending moment causing first yield (i.e. the lowest bending moment at which with elastic behaviour yield stress is attained in tension or compression at one point in the section)
- plastic bending moment (i.e. the bending moment with the section on one side of the neutral axis at the tensile yield stress and on the other side of the neutral axis at compressive yield stress).

for

- nine values of axial force equally spaced between the tensile and compressive yield axial loads of the section.
- angles of applied moment at intervals of 15 degrees from 0 to 345 degrees with reference to the principal axis that is nearest to the beam local y axis.

During calculation cycles the program computes the bending moment at first yield and the plastic bending moment in a beam for the current axial force and angle of applied bending moment by linear interpolation between the values in the "look up" table (both bending moments are of course zero when the axial force equals the tensile or compressive yield force, and the axial force of a beam is limited to values between the tensile and compressive yield forces)

The program initially calculates the forces and bending moments at each end of a beam assuming elastic behaviour. If the net bending moment at the first end is greater than the moment causing first yield then the bending moment is modified according to the formula

$$M_{i+1} = M_y + 0.5(M_i - M_y) \quad \text{equation C}$$

If the bending moment at the first end of a beam is modified, the bending moment at the second end is obtained by using equation B

$$M_2 = \frac{EI \times S'' \theta}{l} + C \times M_{1,mod}$$

If the bending moment at the second end exceeds that at first yield, it is modified in the same way as was the one at the first end, and the bending moment at the first end is obtained by using equation B

$$M_1 = \frac{EI \times S''\theta}{l} + C \times M_{2,\text{mod}}$$

If this bending moment is greater than the moment causing first yield, the whole process is repeated until the bending moments cease to be modified.

Equation C is equivalent to halving the stiffness of a beam at first yield.

Fabric- Stress computation

The warp and weft directions are assumed to be perpendicular. The direct and shear strains are first computed for the warp and weft directions assuming uniform strains over each triangle and the stresses are calculated from the equations

$$\sigma_{xx} = \frac{E_{xx}\varepsilon_{xx} + E_{yy}\nu_{yx}\varepsilon_{yy}}{1 - \nu_{xy}\nu_{yx}} \quad \text{A}$$

$$\sigma_{yy} = \frac{E_{yy}\varepsilon_{yy} + E_{xx}\nu_{xy}\varepsilon_{xx}}{1 - \nu_{xy}\nu_{yx}} \quad \text{B}$$

$$\sigma_{xy} = G\varepsilon_{xy}$$

where x is the warp direction and y is the weft direction.

The principal stresses are then computed. If a triangle is set to take no compression, compressive principal stresses are set to zero.

The forces exerted by the triangle are calculated from the principal stresses.

Equations A and B are obtained by rewriting

$$\varepsilon_{xx} = \frac{\sigma_{xx} - \nu_{yx}\sigma_{yy}}{E_{xx}}$$

and

$$\varepsilon_{yy} = \frac{\sigma_{yy} - \nu_{xy}\sigma_{xx}}{E_{yy}}$$

Poisson's ratio for pure warp stress ν_{yx} is defined in the material table. ν_{yx} , the Poisson's ratio for pure weft stress is calculated from

$$\nu_{yx} = \frac{E_{xx}\nu_{yx}}{E_{yy}}$$

If no shear modulus is specified it is calculated as

$$G = \frac{0.5(E_{xx} + E_{yy})}{2(1 + 0.5(\nu_{xy} + \nu_{yx}))}$$

For isotropic materials where $E_{xx} = E_{yy} = E$ and $\nu_{xy} = \nu_{yx} = \nu$ this is equivalent to

$$G = \frac{E}{2(1 + \nu)}$$

This corresponds to elastic behaviour.

Applied Displacements and Lagrange Multipliers

The basic equations for a linear static analysis are

$$\mathbf{f} = \mathbf{K}\mathbf{u}$$

The applied displacements are applied by using Lagrange multipliers. The basic concept is that the structure matrix can be modified to apply a displacement condition

$$\begin{Bmatrix} \mathbf{f} \\ \tilde{\mathbf{u}} \end{Bmatrix} = \begin{bmatrix} \mathbf{K} & \mathbf{e} \\ \mathbf{e}^T & 0 \end{bmatrix} \begin{Bmatrix} \mathbf{u} \\ \lambda \end{Bmatrix}$$

Where λ are the Lagrange multipliers used to enforce a constraint condition and $\tilde{\mathbf{u}}$ are the applied displacements.

Expanding the matrix equation gives

$$\mathbf{f} = \mathbf{K}\mathbf{u} + \mathbf{e}\lambda$$

$$\tilde{\mathbf{u}} = \mathbf{e}^T\mathbf{u}$$

or

$$\mathbf{u} = \mathbf{K}^{-1}(\mathbf{f} - \mathbf{e}\lambda)$$

$$\tilde{\mathbf{u}} = \mathbf{e}^T(\mathbf{K}^{-1}(\mathbf{f} - \mathbf{e}\lambda))$$

so

$$(\mathbf{e}^T\mathbf{K}^{-1}\mathbf{f} - \tilde{\mathbf{u}}) = (\mathbf{e}^T\mathbf{K}^{-1}\mathbf{e})\lambda$$

$$\hat{\mathbf{f}} = \hat{\mathbf{K}}^{-1}\lambda$$

Solving this equation gives the Lagrange multipliers, which can then be used in

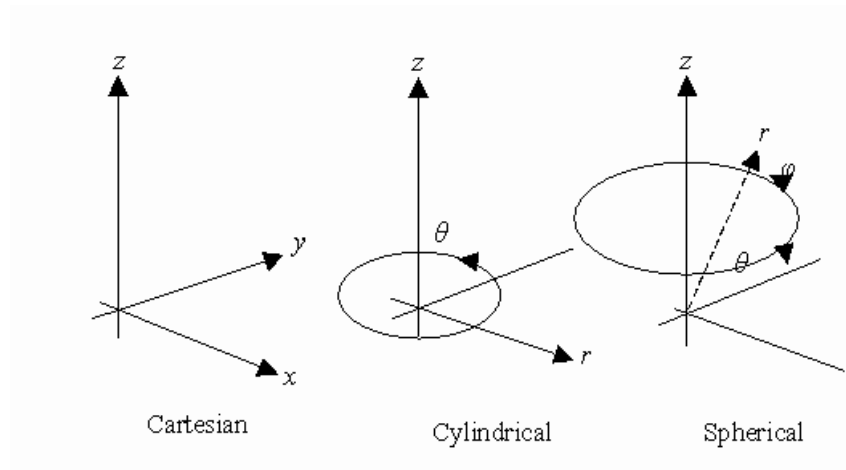
$$\mathbf{u} = \mathbf{K}^{-1}(\mathbf{f} - \mathbf{e}\lambda)$$

to solve for the displacements.

Axes

Axes can be either Cartesian, cylindrical or spherical. The coordinates in these are:

- Cartesian - (x, y, z)
- Cylindrical - (r, θ, z)
- Spherical - (r, θ, ϕ)



An axis is defined by three vectors irrespective of axis type. These define the location and basic orientation. The x axis vector is any vector pointing in the positive x axis direction. The xy plane vector is any vector in the xy plane of the axis that is not parallel with the x axis vector. The axes are then constructed as follows:

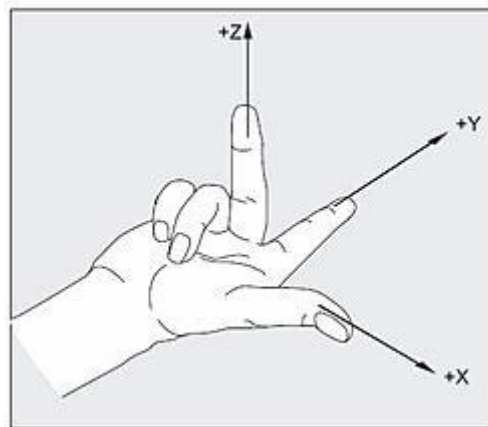
$$\mathbf{x} = \mathbf{x}_{axis} / |\mathbf{x}_{axis}|$$

$$\mathbf{z} = \mathbf{x} \times \mathbf{xy}_{plane} / |\mathbf{x} \times \mathbf{xy}_{plane}|$$

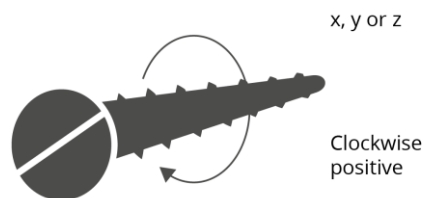
$$\mathbf{y} = \mathbf{z} \times \mathbf{x}$$

The basic axis system is the Global Cartesian axis system, normally referred to as the Global axis system. All other axis systems are located relative to the Global axis system. Global axis directions are generally denoted X , Y and Z to distinguish from other, more general axis directions which use x , y and z .

All the axes systems in GSA are right handed axes systems.



Rotations about the axes follow the right hand screw rule



Composite

Composite Slabs

Composite slabs are a slab supported on steel decking. These can be modelled as a solid slab with adjustment to the in-plane (t_p) and bending (t_b) thickness. For a unit width the area of slab is A concrete (A_c) and steel (A_s) are known as are the second moments of area (I_c and I_s) and the E values (E_c and E_s).

Referring back to the concrete as the primary material the effective area is

$$A_{eff} = A_c + (E_s/E_c)A_s$$

And the effective thickness (in-plane) is

$$t_p = \frac{A_{eff}}{A} = \frac{A_c + (E_s/E_c)A_s}{A}$$

Give the centroid of the concrete (z_c) and steel decking (z_s) the centroid of the composite section is then

$$z_{eff} = \frac{A_c z_c + (E_s/E_c)A_s z_s}{A_c + (E_s/E_c)A_s}$$

and the effective second moment of area (I_{eff}) is

$$I_{eff} = \left[I_c + A_c (z_c - z_{eff})^2 \right] + (E_s / E_c) \left[I_s + A_s (z_s - z_{eff})^2 \right]$$

And the effective thickness in bending is

$$t_b = \sqrt[3]{\frac{I_{eff}}{I}} = \sqrt[3]{\frac{\left[I_c + A_c (z_c - z_{eff})^2 \right] + (E_s / E_c) \left[I_s + A_s (z_s - z_{eff})^2 \right]}{I}}$$

Effective Elastic Properties

In order to simplify calculations it is possible to determine effective elastic properties of a section. The simplest of these is the area. Consider a section with both concrete and steel with areas A_c and A_s respectively. The axial stiffnesses are

$$k_c = \frac{A_c E_c}{l}$$

$$k_s = \frac{A_s E_s}{l}$$

And the total stiffness is then

$$k = \frac{A_c E_c}{l} + \frac{A_s E_s}{l}$$

To simplify calculation we can choose a reference material. So for concrete as a reference material

$$\frac{A_{eff} E_c}{l} = \frac{A_c E_c}{l} + \frac{A_s E_s}{l}$$

or

$$A_{eff} = A_c + A_s \frac{E_s}{E_c}$$

More generally for a collection of components with a reference section

$$A_{eff} = \frac{\sum_i (A_i E_i)}{E_{ref}}$$

For a section made of multiple components the effective centroid is defined as

$$c_{eff} = \frac{\sum_i \{A_i (c_i - c_{ref})\}}{A_{eff}}$$

As for axial properties effective bending properties can be defined (allowing for the different centroids) as

$$I_{eff} = \frac{\sum_i \{ (I_i + A_i (c_i - c_{eff})^2) E_i \}}{E_{ref}}$$

Condition Number

Ill conditioning arises while solving linear equations of the type

$$\mathbf{f} = \mathbf{K} \mathbf{x}$$

for given loads \mathbf{f} and stiffness \mathbf{K} in (say) linear static analysis, approximations are introduced in the solution because all calculations are carried out in finite precision arithmetic. This becomes important when \mathbf{K} is ill-conditioned because there is a possibility of these approximations leading to large errors in the displacements. The extent of these errors can be quantified by the 'condition number' of the stiffness matrix.

The condition number of a matrix (with respect to inversion) measures worst-case of changes in $\{\mathbf{x}\}$ corresponding to small changes in \mathbf{K} or \mathbf{f} . It can be calculated using the product of norm of the matrix times the norm of its inverse.

$$\kappa(\mathbf{K}) = \|\mathbf{K}\| \cdot \|\mathbf{K}^{-1}\|$$

where $\|\cdot\|$ is a subordinate matrix norm.

If \mathbf{K} is a symmetric matrix, the condition number $\kappa(\mathbf{K})$ can be shown to be the ratio of its maximum and minimum eigenvalues λ_{\max} and λ_{\min} .

$$\kappa(\mathbf{K}) = \frac{\lambda_{\max}}{\lambda_{\min}}$$

The minimum value of $\kappa(\mathbf{K})$ is 1 and the maximum value is infinity. If the condition number is small, the computed solution \mathbf{x} is reliable (i.e. a reliable approximation to the true solution of $\mathbf{f} = \mathbf{K} \mathbf{x}$). If the condition number is large, (i.e. if the matrix is almost singular) the results cannot be trusted.

GSA computes a lower bound approximation to the 1-norm condition number of \mathbf{K} and this is reported as part of the solver output. This can be used to evaluate the accuracy of the solution both qualitatively and quantitatively. The (qualitative) rule of thumb for accuracy is $n - \log \kappa$ - the number of digits of accuracy in \mathbf{x} is

$$n = 16 - \log \kappa$$

In general any stiffness matrix with condition number above 10^{15} can produce results with no accuracy at all. Any results produced from matrices with condition number greater than 10^{10} must be treated with caution.

Where a model is ill-conditioned, Model Stability analysis can help detect the causes of ill conditioning.

For a given condition number, we can also compute the maximum relative error in \mathbf{x} . The maximum relative error in \mathbf{x} is defined as the maximum ratio of norms of error in \mathbf{x} to \mathbf{x} , i.e.

$$e_{\max} = \frac{|\Delta \mathbf{x}|}{|\mathbf{x}|}$$

Given a matrix \mathbf{K} with condition number κ , the maximum relative error in \mathbf{x} when solving $\mathbf{f} = \mathbf{K} \mathbf{x}$ is

$$\frac{2\varepsilon\kappa}{1 - \varepsilon\kappa}$$

where ε is the constant 'unit-roundoff' and is equal to 1.11e-16 for double precision floating point numbers. The maximum relative error is computed and reported as part of solver output. Ideally, this should be small ($\ll 1$), since a small relative error indicates a reliable solution but as $\kappa \rightarrow 1/\varepsilon$, the relative error grows rapidly.

GSA calculates the condition number using Higham and Tisseur's block 1-Norm condition number estimation algorithm.

These are reported in the Analysis Details output.

Constraint & Constraint Equations

Constraint equations for the basis of the different constraint types in GSA:

- Joints
- Rigid constraints
- Constraint equations
- Tied interfaces

Constraint Equation

The form of a constraint equation is

$$u_s = \sum a_i u_{m,i}$$

and is used to tie degrees of freedom in the model.

Joint

For a joint this becomes a set of equations for the linked degrees of freedom of the form

$$u_s = u_m$$

Rigid Constraints

For a rigid constraint there are a set of constraint equations which respect the geometry of the constraint. So for a single slave node the constraint equations are

$$\begin{Bmatrix} u_s \\ v_s \\ w_s \\ \theta_{us} \\ \theta_{vs} \\ \theta_{ws} \end{Bmatrix} = \begin{bmatrix} 1 & & & z & -y \\ & 1 & & -z & x \\ & & 1 & y & -x \\ & & & \delta & \\ & & & & \delta \\ & & & & \delta \end{bmatrix} \begin{Bmatrix} u_m \\ v_m \\ w_m \\ \theta_{um} \\ \theta_{vm} \\ \theta_{wm} \end{Bmatrix}$$

where the δ terms are 1 for a fixed and 0 for a pinned rigid constraint.

Different terms in the matrix are dropped for reduced constraint types. The two most common special types are plane and plate constraints with equations (for the xy case)

$$\begin{Bmatrix} u_s \\ v_s \\ \theta_{ws} \end{Bmatrix} = \begin{bmatrix} 1 & & -y \\ & 1 & x \\ & & \delta \end{bmatrix} \begin{Bmatrix} u_m \\ v_m \\ \theta_{wm} \end{Bmatrix}$$

for an xy plane constraint

$$\begin{Bmatrix} w_s \\ \theta_{us} \\ \theta_{vs} \end{Bmatrix} = \begin{bmatrix} 1 & y & -x \\ & \delta & \\ & & \delta \end{bmatrix} \begin{Bmatrix} w_m \\ \theta_{um} \\ \theta_{vm} \end{Bmatrix}$$

for a z plate constraint.

Tied interfaces

Tied interfaces are composed of master and slave surfaces. Internally these are broken down to nodes on the slave side and element (faces) on the master side. The nodes on the slave side are connected to the adjacent master face via a set of constraint equations.

The (r,s) coordinates of the nodes relative to the master face are established and then the shape functions are used to construct a set of constraint equations

$$u_s(r, s) = \sum_i h_i u_{m,i}$$

In the case of a quad-4 face this expands to

$$u_s(r, s) = \frac{1}{4}(1-r)(1-s)u_{m,1} + \frac{1}{4}(1+r)(1-s)u_{m,2} + \frac{1}{4}(1+r)(1+s)u_{m,3} + \frac{1}{4}(1-r)(1+s)u_{m,4}$$

which forms the constraint equation. This is repeated for all the displacement directions.

The special case is the drilling degree of freedom. As the 2D elements have either no drilling freedom or one which can work quite locally. For this degree of freedom the rotation is linked to the translations of the 2D element. If the node is internal to the element base the rotation of the element as a whole. If the node is on the edge use the rotation of just that edge.

For each element node define a vector \mathbf{V}_i from the slave node position to the node in the plane of the element. Let

$$l_i = |\mathbf{v}_i|$$

$$\tan \alpha_i = \frac{v_{i,y}}{v_{i,x}}$$

The displacement at the centre of the 2D element is

$$u_c = \sum_i h_i u_i$$

Then the rotation of the node at a distance l_i is and angle α_i

$$\theta_i = \frac{-(u_{i,x} - u_{c,x}) \sin \alpha_i + (u_{i,y} - u_{c,y}) \cos \alpha_i}{l_i}$$

So rotation at (r,s) is

$$\theta(r, s) = \sum_i h_i \frac{-(u_{i,x} - u_{c,x}) \sin \alpha_i + (u_{i,y} - u_{c,y}) \cos \alpha_i}{l_i}$$

Or expanding

$$\theta(r, s) = \sum_i h_i \frac{-u_{i,x} \sin \alpha_i + u_{i,y} \cos \alpha_i}{l_i} - \sum_i h_i \frac{-u_{c,x} \sin \alpha_i + u_{c,y} \cos \alpha_i}{l_i}$$

$$\theta(r, s) = \sum_i h_i \frac{-u_{i,x} \sin \alpha_i + u_{i,y} \cos \alpha_i}{l_i} - \sum_i h_i \frac{-\sum_j (h_j u_{j,x}) \sin \alpha_i + \sum_j (h_j u_{j,y}) \cos \alpha_i}{l_i}$$

Direction Cosines

Direction cosines contain information that allows transformation between local and global axis sets. Given a set of orthogonal unit axis vectors the direction cosine array is defined as

$$D = [x \mid y \mid z]$$

Any vector or tensor can then be transformed from local to global through

$$v_g = D v_l$$

$$t_g = D t_l D^T$$

The inverse transformation uses the transpose of the direction cosine array

$$v_l = D^T v_g$$

$$t_l = D^T t_g D$$

Dynamic Response Analysis

Harmonic Analysis

Harmonic analysis is used to calculate the elastic structure responses to harmonic (sinusoidally varying) loads at steady state. This is done using modal superposition.

The dynamic equation of motion is:

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{p}\sin(\omega t)$$

Where \mathbf{p} represents the spatial distribution of load and ω the time variation.

From the mode shape results of a modal dynamic analysis, the nodal displacements, velocities and accelerations can be expressed as

$$\mathbf{u} = \Phi \mathbf{q}$$

$$\dot{\mathbf{u}} = \Phi \dot{\mathbf{q}}$$

$$\ddot{\mathbf{u}} = \Phi \ddot{\mathbf{q}}$$

where \mathbf{q} , $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$ are the displacement, velocity and acceleration in modal (generalized) coordinates, for the m modes analysed.

Substituting these in the original equation gives

$$\mathbf{M}\Phi\ddot{\mathbf{q}} + \mathbf{C}\Phi\dot{\mathbf{q}} + \mathbf{K}\Phi\mathbf{q} = \mathbf{p}\sin(\omega t)$$

Pre-multiplying each term in this equation by the transpose of the mode shape gives

$$\Phi^T \mathbf{M}\Phi\ddot{\mathbf{q}} + \Phi^T \mathbf{C}\Phi\dot{\mathbf{q}} + \Phi^T \mathbf{K}\Phi\mathbf{q} = \Phi^T \mathbf{p}\sin(\omega t)$$

According to the orthogonality relationship of the mode shapes to the mass matrix and the stiffness matrix and also assuming the mode shapes are also orthogonal to the damping matrix (e.g. Rayleigh damping), this equation can be replaced by a set of m uncoupled dynamic equations of motion as shown below.

$$\varphi_i^T \mathbf{M}\varphi_i \ddot{q}_i + \varphi_i^T \mathbf{C}\varphi_i \dot{q}_i + \varphi_i^T \mathbf{K}\varphi_i q_i = \varphi_i^T \mathbf{p}\sin(\omega t)$$

Setting

$$\begin{aligned}\hat{m}_i &= \varphi_i^T \mathbf{M} \varphi_i \\ \hat{k}_i &= \varphi_i^T \mathbf{K} \varphi_i \\ \hat{c}_i &= \varphi_i^T \mathbf{C} \varphi_i \\ \hat{p}_i &= \varphi_i^T \mathbf{p}\end{aligned}$$

Then the uncoupled equations can be expressed in a general form as follows

$$\hat{m}_i \ddot{q}_i + \hat{c}_i \dot{q}_i + \hat{k}_i q_i = \hat{p}_i \sin(\omega t)$$

where all the terms are scalars. Solving this equation is equivalent to solving a single degree of freedom problem.

For the single degree of freedom problem subjected to harmonic load, the dynamic magnification factors μ of the displacement for mode i in complex number notation is

$$\mu_i = \left[\left(1 - \left(\frac{\omega}{\omega_i} \right)^2 + i 2\xi \left(\frac{\omega}{\omega_i} \right) \right) \right]^{-1} = \mu_{i,\Re} - i \mu_{i,\Im}$$

where

$$\begin{aligned}\mu_{\Re} &= \frac{A}{A^2 + B^2} & \mu_{\Im} &= \frac{B}{A^2 + B^2} \\ A &= 1 - \left(\frac{\omega}{\omega_i} \right)^2, & b &= 2\xi \left(\frac{\omega}{\omega_i} \right)\end{aligned}$$

and ω_i is the natural frequency of mode i .

The maximum displacement, velocity & acceleration of mode i in the modal coordinates are

$$\begin{aligned}q_i &= \mu \frac{\hat{p}_i}{\hat{k}_i} = \mu \frac{\hat{p}_i}{\hat{m}_i \omega_i^2} \\ \dot{q}_i &= q_i \omega = \mu \frac{\hat{p}_i \omega}{\hat{k}_i} = \mu \frac{\hat{p}_i \omega}{\hat{m}_i \omega_i^2} \\ \ddot{q}_i &= q_i \omega^2 = \mu \frac{\hat{p}_i \omega^2}{\hat{k}_i} = \mu \frac{\hat{p}_i \omega^2}{\hat{m}_i \omega_i^2}\end{aligned}$$

Substituting gives the maximum actual nodal displacements, velocities & accelerations at the steady state of the forced vibration as

$$\mathbf{u} = \sum_{i=1}^m \varphi_i q_i \quad \dot{\mathbf{u}} = \sum_{i=1}^m \varphi_i \dot{q}_i \quad \ddot{\mathbf{u}} = \sum_{i=1}^m \varphi_i \ddot{q}_i$$

After obtaining the maximum nodal displacements, the element forces and moments etc can be calculated as in static analysis.

Periodic Load Analysis

GSA periodic load analysis is to calculate the maximum elastic structure responses to generic periodic loads at steady state. Modal superposition method is used in GSA periodic load analysis.

The dynamic equation of motion subjected to periodic loads is

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{p} f(t)$$

Where $f(t)$ is a harmonic load function. Using a Fourier Series, the periodic function of time can be expressed as a number of sine functions

$$f(t) = \sum_{h=1}^H r_h \sin\left(\frac{2\pi h}{T} t\right)$$

where r_h are the Fourier coefficients (or dynamic load factor) defined by the user and T is the period of the periodic load frequency and H is the number of Fourier (harmonic) terms to be considered.

Substituting in the first equation we can rewrite as a number of dynamic equations of motion subjected to harmonic loads:

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{p} r_h \sin\left(\frac{2\pi h}{T} t\right)$$

The maximum responses of this can be solved using harmonic analysis for each of the harmonic loads ($h = 1, 2, \dots$) then the maximum responses from the periodic loads can be calculated using square root sum of the squares (SRSS)

$$R_{\max} = \sqrt{\sum_{h=1}^H R_{h,\max}^2}$$

Linear Time-history Analysis

Linear time history analysis is used to calculate the transient linear structure responses to dynamic loads or base acceleration using modal superposition.

The dynamic equation of motion of structure subjected to dynamic loads is

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{p} f(t)$$

If the excitation is base acceleration

$$\mathbf{p} = \mathbf{M}\mathbf{v}$$

where \mathbf{v} is an influence vector that represents the displacement of the masses resulting from static application of a unit base displacement defined by the base excitation direction and the force due to the base acceleration is

$$f(t) = \ddot{u}_g(t)$$

To use the results (mode shapes) from modal dynamic analysis, the nodal displacements, velocities and accelerations can be expressed in modal coordinates as

$$\begin{aligned}\mathbf{u} &= \Phi \mathbf{q} \\ \dot{\mathbf{u}} &= \Phi \dot{\mathbf{q}} \\ \ddot{\mathbf{u}} &= \Phi \ddot{\mathbf{q}}\end{aligned}$$

Then setting

$$\begin{aligned}\hat{m}_i &= \varphi_i^T \mathbf{M} \varphi_i \\ \hat{k}_i &= \varphi_i^T \mathbf{K} \varphi_i \\ \hat{c}_i &= \varphi_i^T \mathbf{C} \varphi_i \\ \hat{p}_i &= \varphi_i^T \mathbf{p}\end{aligned}$$

gives

$$\hat{m} \ddot{\mathbf{q}} + \hat{c} \dot{\mathbf{q}} + \hat{k} \mathbf{q} = \hat{\mathbf{p}} f(t)$$

This gives a single degree of freedom problem that can be solved using any of the direct numerical analysis methods such as Newmark or central differences (Newmark is used in GSA). There are m such equations that are corresponding to each of the modes from the modal dynamic analysis. Superimposing the responses from each of the one degree of freedom problem the total responses of the structure can be calculated from

$$\mathbf{u} = \sum_{i=1}^m \varphi_i q_i \quad \dot{\mathbf{u}} = \sum_{i=1}^m \varphi_i \dot{q}_i \quad \ddot{\mathbf{u}} = \sum_{i=1}^m \varphi_i \ddot{q}_i$$

Footfall Analysis

Footfall analysis (or in full, footfall induced vibration analysis) is used to calculate the elastic vertical nodal responses (acceleration, velocity, response factor etc.) of structures to human footfall loads (excitations). The human footfall loads $f(t)$ are taken as periodic loads. Using to Fourier Series, the period footfall loads can be expressed as:

$$f(t) = G \left[1 + \sum_{h=1}^H r_h \sin\left(\frac{2\pi h}{T} t\right) \right]$$

where G is the body weight of the individual, and r_h are the Fourier coefficients (or dynamic load factor), the actual values of dynamic load factors can be found from reference [24](#), [25](#) and [26](#) in the bibliography, T is the period of the footfall (inverse of walking frequency) and H the number of Fourier (harmonic) terms to be considered, 4 is used for walking on floor using CCIP-016 method, 3 is used for walking on floor using SCI method and 2 is used for walking on stairs.

After subtracting the static weight of the individual (since it does not vary with time and does not induce any dynamic response), the dynamic part of the footfall loads are the sum of a number of harmonic loads

$$f(t) = G \sum_{h=1}^H r_h \sin\left(\frac{2\pi h}{T} t\right)$$

There are two distinctive responses from the footfall excitation, the resonant (steady state) and transient. If the minimum natural frequency of a structure is higher than 4 times the highest walking frequency (see reference 24), the resonant response is normally not excited since the natural frequencies of the structure are so far from the walking (excitation) frequency, therefore the transient response is normally in control, otherwise, the resonant response is probably in control. Both resonant (steady state) and transient analyses are considered in GSA footfall analysis, so the maximum responses will always be captured.

Resonant response analysis

As footfall loads are composed of a number of harmonic loads (components), harmonic analysis is used to get the responses for each of the harmonic components of footfall loads and then to combined them to get the total responses. From one of the harmonic components (h) of the footfall loads in equation above and the given walking frequency ($1/T$), the following dynamic equation of motion can be obtained

$$\mathbf{M}\ddot{\mathbf{u}}_h + \mathbf{C}\dot{\mathbf{u}}_h + \mathbf{K}\mathbf{u}_h = \delta_k G r_h \sin\left(\frac{2\pi h}{T} t\right)$$

where δ_k is a unit vector used to define the location of the harmonic load. All the components in this vector are zero except the term that corresponds to the vertical direction of the node subjected footfall load.

Since the number of footfalls is limited and the full resonant response from the equation above may not always be achieved, a reduction factor $\rho_{h,m}$ for the dynamic magnification factors μ is needed to account for this non-full resonant response. The reduction factor can be calculated from

$$\rho_{h,m} = 1 - e^{-2\pi\zeta_m N}$$

Where ζ_m is damping ratio of mode m and $N = 0.55hW$ with h the harmonic load number and W the number of footfalls.

Applying this reduction factor to the dynamic magnification factors (μ) in Harmonic Analysis, this equation can be solved using the method described in Harmonic Analysis Theory section. Repeating this analysis, the responses from the other harmonic loads of the footfall can also be obtained. The interested results from this analysis are the total vertical acceleration and the response factor from all harmonic loads of the footfall. The total vertical acceleration is taken as the square root of the sum of squares of the accelerations from each of the harmonic analyses. The response factor for each of the harmonic loads is the ratio of the nodal acceleration to the base curve of the Root Mean Square acceleration given in [reference 25](#) as shown below. This total response factor is then taken as the square root of the sum of squares of the response

factors from each of the harmonic loads. According to this, the total acceleration and response factor can be calculated from

$$a_i = \sqrt{\sum_{h=1}^H \ddot{u}_{i,h}^2}$$

$$f_i = \sqrt{\sum_{h=1}^H f_{i,h}^2} = \sqrt{\sum_{h=1}^H \left(\frac{\ddot{u}_{i,h}^2 w_i}{0.005\sqrt{2}} \right)}$$

where

a_i is the maximum acceleration at node i

$\ddot{u}_{i,h}$ is the maximum acceleration at node i by the excitation of harmonic load h

H is the number of harmonic components of the footfall loads considered in the analysis

f_i is the response factor at node i

$f_{i,h}$ is the response factor at node i by the excitation of harmonic load h

w_i is the frequency weighting factor and it is a function of frequency

For standard weighting factors see Table 3 of BS6841.

Transient response analysis

The transient response of structures to footfall forces is characterised by an initial peak velocity followed by a decaying vibration at the natural frequency of the structure until the next footfall. As the natural frequencies of the structure considered in this analysis is much higher the highest walking frequency, there is no tendency for the response to build up over time as it does in resonant response analysis. The maximum response will be at the beginning of each footfall. Each footfall is considered as an impulse to the structure, according to [references 35 & 29](#), the design impulse can be calculated from

When walking on floor (Concrete Centre/Arup method)

$$I_{des,m} = 54 \frac{f^{1.43}}{f_m^{1.3}}$$

When walking on floor (SCI P354 method)

$$I_{des,m} = 60 \frac{f^{1.43}}{f_m^{1.3}} \frac{Q}{700}$$

When walking on stairs (Concrete Centre/Arup method)

$$I_{des,m} = 54 \frac{150}{f_m}$$

When walking on stairs (SCI P354 method)

$$I_{des,m} = 0$$

where

$I_{des,m}$ is the design impulse for mode m in NS

f is the walking frequency in Hz

f_m is the natural frequency of the structure in mode m in Hz

Q is the weight of the walker in N

For this impulse, the peak velocity in each mode is given by

$$\hat{v}_m = u_{e,m} u_{r,m} \frac{I_{des,m}}{\hat{m}_m}$$

and the peak acceleration in each mode is given by

$$\hat{a}_m = 2\pi f_m \hat{v}_m = 2\pi f_m u_{e,m} u_{r,m} \frac{I_{des,m}}{\hat{m}_m}$$

where

\hat{v}_m is the peak velocity in mode m by the footfall impulse

\hat{a}_m is the peak acceleration in mode m by the footfall impulse

$u_{e,m}, u_{r,m}$ are the vertical displacements at the excitation and response nodes respectively in mode m

\hat{m}_m is the modal mass in mode m

The variation of the velocity with time of each mode is given by

$$v_m(t) = \hat{v}_m e^{-2\pi\xi f_m t} \sin(2\pi f_m t)$$

and the variation of the acceleration with time of each mode is given:

$$a_m(t) = \hat{a}_m e^{-2\pi\xi f_m t} \sin(2\pi f_m t)$$

where ξ is the damping ratio associated with mode m

The final velocity and acceleration at the response node are the sum of the velocities and accelerations of all the modes (M) that are considered

$$v(t) = \sum_{m=1}^M v_m(t)$$

$$a(t) = \sum_{m=1}^M a_m(t)$$

This gives the peak velocity and peak acceleration. The root mean square velocity and root mean square acceleration can be calculated from the period of the footfall

$$v_{RMS} = \sqrt{\frac{1}{T} \int_0^T v^2(t) dt}$$

$$a_{RMS} = \sqrt{\frac{1}{T} \int_0^T a^2(t) dt}$$

The response factor at time t (t is from 0 to T and T is the period of the footfall loads) can be calculated from

$$f_R(t) = \frac{1}{0.005} \sum_{m=1}^M a_m(t) w_m$$

where

w_m is the frequency weighting factor corresponding to the frequency of mode m

The final transient response factor, based on the root mean square principle, is given by

$$f_R = \sqrt{\frac{1}{T} \int_0^T f_R^2(t) dt}$$

Element axes

The orientation of elements depends on the element type. These are represented by the direction cosines based on the element x, y and z axis directions.

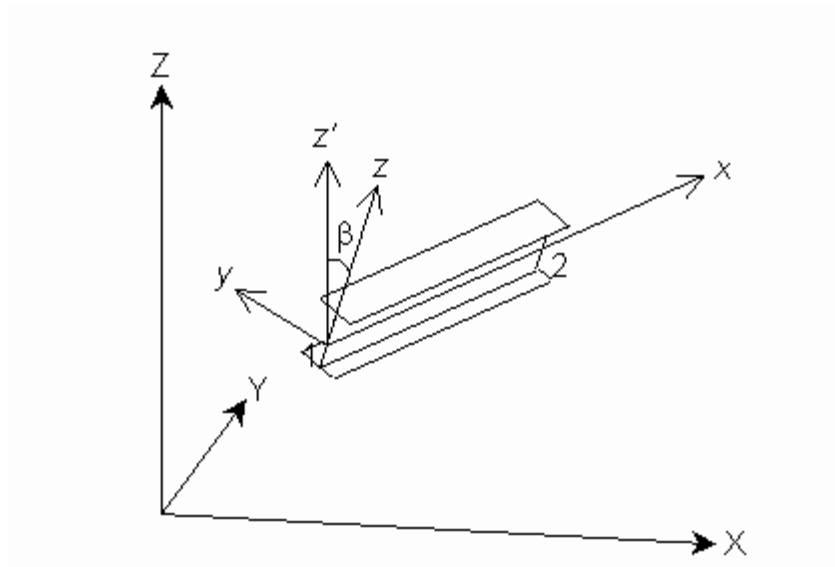
Section elements

Section elements include beams, bars, rods, ties and struts are defined by two nodes locating the ends of the element. The x axis of the element is along the axis of the element (taking account of any offsets) from the first topology item to the second.

The definition of the element y and z axes then depends on the element's orientation, verticality, orientation node and orientation angle. The element is considered vertical in GSA if the element is within the 'vertical element tolerance'.

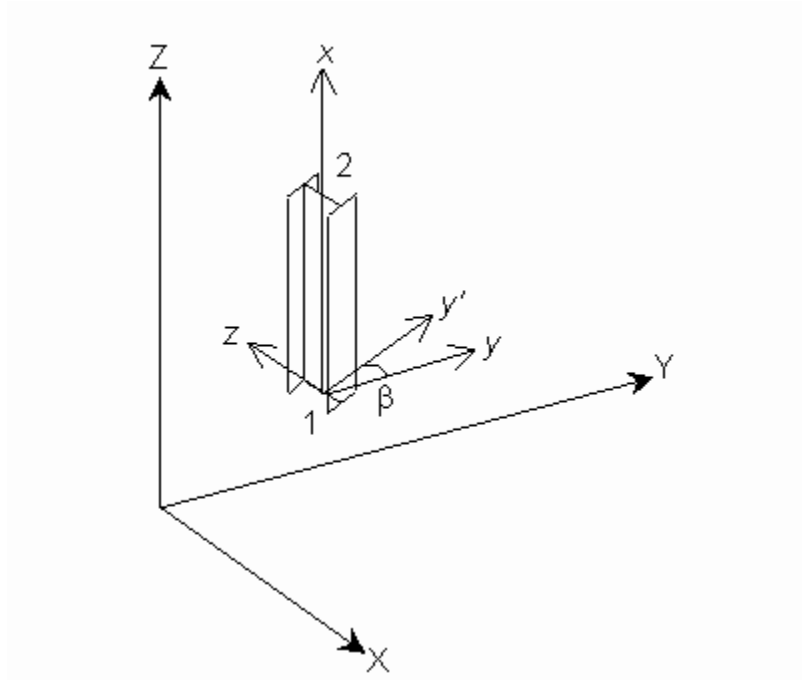
Non-vertical elements

If an orientation node is not specified, the element z axis of a non-vertical element defaults to lying in the vertical plane through the element and is directed in the positive sense of the global Z direction. The element y axis is orthogonal to the element z and x axes. The element y and z axes may be rotated out of this default position by the orientation angle.



Vertical elements

If an orientation node is not specified, the element y axis of a vertical element defaults to being parallel to and is directed in the positive sense of the global Y axis. The element z axis is orthogonal to the element x and y axes. The element y and z axes may be rotated out of this default position by the orientation angle.



Orientation node

If an orientation node is specified, the element x-y plane is defined by the element x axis and a vector from the first topology position to the orientation node, such that the node has a positive y coordinate. The element z axis is orthogonal to the element x and y axes. Specifying an orientation node overrides the “vertical element” and “non-vertical element” definitions described above. The element y and z axes may be rotated out of this default position by the orientation angle.

Orientation angle

The element y and z axes are rotated from their default positions about the element x axis by the orientation angle in the direction following the right hand screw rule. This occurs regardless of whether or not the element is vertical and of whether or not an orientation node is specified.

Spring elements

The definition of the element axes for a spring depends on the spring element. For a two-noded element with non-zero length the definition of the local axis is the same as for a beam element. If the element is a ground spring or has zero length the axis system is set in the spring property. If the axis system is ‘local’ then the rules for orientating section elements apply. If the axis system is ‘global’ or user defined, these are used and the topology, orientation node and orientation angle are ignored. User defined axes can be Cartesian, cylindrical or spherical.

Grounded spring and mass elements

The definition of the element axes for a grounded spring or mass depends on the axis system set in the spring or mass property. If the axis system is ‘global’ or user defined, these are used. If

the axis system is 'local' the element axis is taken as global. User defined axes can be Cartesian, cylindrical or spherical.

Cable elements

Cable elements act only in the axial direction, so only the x axis is defined following the same definition as the x axis of a beam element.

Link elements

The local axes of link elements are the same as those of the master node.

2D element axes

2D element local axes may be defined either by reference to an axis set or topologically. This is determined by the axis system defined in the 2D element property. If the axis system is 'global' or user defined then the specified axis set is used. If the axis system is 'local' then the topological definition is applied. User defined axes can be Cartesian, cylindrical or spherical.

Typically defining 2D element local axes by reference to an axis set results in more consistent local axes in the mesh.

The local axes for flat 2D elements are chosen so that the plane of the element is the local x-y plane.

The normal to the element is defined as

$$n = (c_3 - c_1) \times (c_4 - c_2)$$

Where c_i is the coordinates on a point on the element, i.e. the coordinates of the node, i , plus any offset, o_i , at that topology position.

$$c_i = c_{0,i} + o_i$$

2D element axes defined by axis set

If the 2D element property axis is set to other than 'local' then the specified axis system is projected on to the element. For Cartesian axes the x axis of the axis set is projected onto the element

$$y = (n \times x_{axis}) / |n \times x_{axis}|$$

$$x = y \times n$$

$$z = n$$

The exception to this rule is when the x axis of the axis set is within 1° of the element normal in which case a vector for an interim y axis is defined as

$$y = (n \times z_{axis}) / |n \times z_{axis}|$$

$$x = y \times n$$

$$z = n$$

This axis set is then rotated about the element normal equivalent to an orientation angle of 90°.

For Cylindrical and Spherical axes the z axis of the axis set is projected on to the element to become the local y axis.

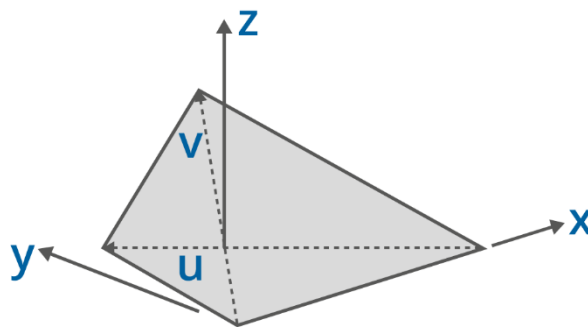
$$x = (z_{axis} \times n) / |z_{axis} \times n|$$

$$y = n \times x$$

$$z = n$$

Topological definition of 2D element axes

If the 2D element property axis is set to 'local' the local x and y axes are based on the topology of the element.



$$u = c_4 - c_2$$

$$v = c_3 - c_1$$

$$n = u \times v$$

$$x = (c_2 - c_1) / |c_2 - c_1|$$

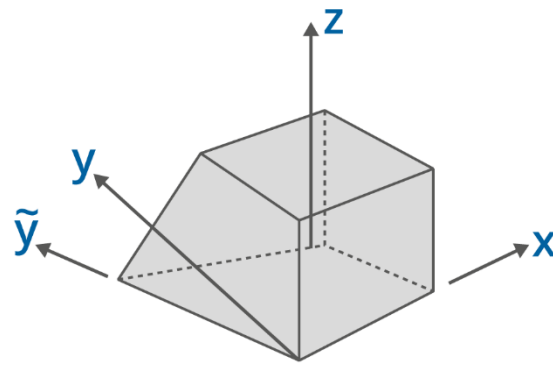
$$y = n \times x$$

$$z = n$$

If an orientation angle is defined these axes are rotated by the orientation angle in a positive direction about the element z axis.

3D element axes

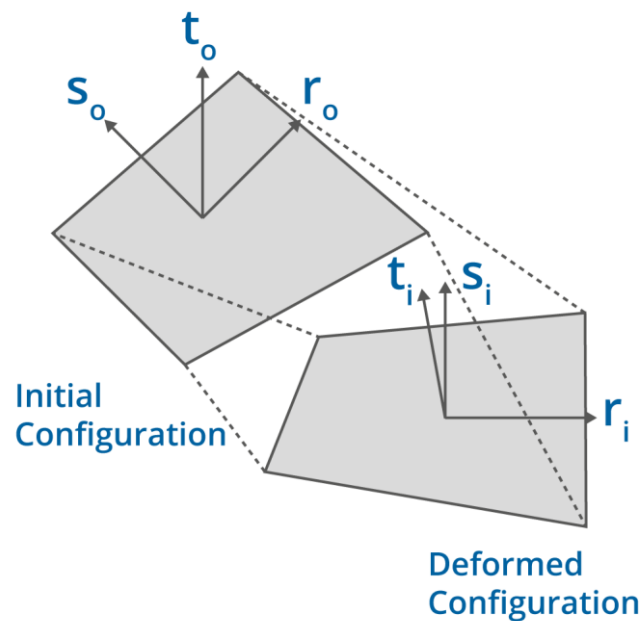
3D element local axes may be defined either by reference to an axis set or topologically.



Updated element axes

In a geometrically non-linear analysis the axes of the element must deform with the element. This has to resolve the difference between the original (undeformed) configuration and the current (deformed) configuration.

For 1D elements the deformed direction cosines can be represented by a new x vector based on the deformed positions of the ends of the element and the average rotation of the element about its x-axis.



For 2D elements the undeformed configuration can be represented by direction cosines based on the (r_0, s_0, t_0) axes.

$$D_0 = [r_0 | s_0 | t_0]$$

The deformed configuration at i can be represented by direction cosines based on the deformed (r_i, s_i, t_i) axes.

$$D_i = [r_i | s_i | t_i]$$

The base direction cosines D_{e0} can then be updated using

$$D_{ei} = D_{e0} D_0^T D_i$$

Elements

Bar and Rod Elements

The bar element stiffness is

$$K = \frac{AE}{l} \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ & & & 1 & 0 & 0 \\ & & & & 0 & 0 \\ & & & & & 0 \end{bmatrix}$$

The mass matrix is

$$M = \rho Al \begin{bmatrix} \frac{1}{3} & 0 & 0 & \frac{1}{6} & 0 & 0 \\ & \frac{1}{3} & 0 & 0 & \frac{1}{6} & 0 \\ & & \frac{1}{3} & 0 & 0 & \frac{1}{6} \\ & & & \frac{1}{3} & 0 & 0 \\ & & & & \frac{1}{3} & 0 \\ & & & & & \frac{1}{3} \end{bmatrix}$$

And the geometric stiffness is

$$K_g = \frac{F_x}{l} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ & 1 & 0 & 0 & -1 & 0 \\ & & 1 & 0 & 0 & -1 \\ & & & 0 & 0 & 0 \\ & & & & 1 & 0 \\ & & & & & 1 \end{bmatrix}$$

The rod element stiffness is

$$K = \begin{bmatrix} \frac{AE}{l} & 0 & 0 & 0 & 0 & 0 & -\frac{AE}{l} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{GJ}{l} & 0 & 0 & 0 & 0 & 0 & -\frac{GJ}{l} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{AE}{l} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{GJ}{l} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Cable, Tie and Strut Elements

These elements are variants on the bar element. Cable and tie elements can carry only tensile forces, while strut elements can carry only compressive forces. The stiffness matrix for ties and struts is identical to that for a bar elements. The non-linear aspect being part of the solver, solution process. The stiffness of cable elements is similar to a bar element but the AE/l is replaced by a stiffness term where the cable stiffness is

$$k_{cable} \equiv AE$$

Beam Elements

The beam element stiffness is

$$\mathbf{K} = E \begin{bmatrix}
 \frac{A}{l} & 0 & 0 & 0 & 0 & 0 & -\frac{A}{l} & 0 & 0 & 0 & 0 & 0 \\
 & \frac{12I_{yy}}{l^3} & 0 & 0 & 0 & \frac{6I_{yy}}{l^2} & 0 & -\frac{12I_{yy}}{l^3} & 0 & 0 & 0 & \frac{6I_{yy}}{l^2} \\
 & & \frac{12I_{zz}}{l^3} & 0 & -\frac{6I_{zz}}{l^2} & 0 & 0 & 0 & -\frac{12I_{zz}}{l^3} & 0 & -\frac{6I_{zz}}{l^2} & 0 \\
 & & & \frac{GJ}{El} & 0 & 0 & 0 & 0 & 0 & -\frac{GJ}{El} & 0 & 0 \\
 & & & & \frac{4I_{zz}}{l} & 0 & 0 & 0 & \frac{6I_{zz}}{l^2} & 0 & \frac{2I_{zz}}{l} & 0 \\
 & & & & & \frac{4I_{yy}}{l} & 0 & \frac{6I_{yy}}{l^2} & 0 & 0 & 0 & \frac{2I_{yy}}{l} \\
 & & & & & & \frac{A}{l} & 0 & 0 & 0 & 0 & 0 \\
 & & & & & & & \frac{12I_{yy}}{l^3} & 0 & 0 & 0 & -\frac{6I_{yy}}{l^2} \\
 & & & & & & & & \frac{12I_{zz}}{l^3} & 0 & \frac{6I_{zz}}{l^2} & 0 \\
 & & & & & & & & & \frac{GJ}{El} & 0 & 0 \\
 & & & & & & & & & & \frac{4I_{zz}}{l} & 0 \\
 & & & & & & & & & & & \frac{4I_{yy}}{l}
 \end{bmatrix}$$

These are modified for a shear beam as follows

$$\alpha = 12 \frac{EI_{ii}}{l^2 GA_s}, \quad A_s = Ak_j$$

$$\frac{2EI}{l} \rightarrow \left(\frac{2-\alpha}{1+\alpha} \right) \frac{EI}{l}, \quad \frac{6EI}{l^2} \rightarrow \left(\frac{6}{1+\alpha} \right) \frac{EI}{l^2}$$

$$\frac{4EI}{l} \rightarrow \left(\frac{4-\alpha}{1+\alpha} \right) \frac{EI}{l}, \quad \frac{12EI}{l^3} \rightarrow \left(\frac{12}{1+\alpha} \right) \frac{EI}{l^3}$$

The mass matrix is

$$M = \frac{\rho A l}{420} \begin{bmatrix} 140 & 0 & 0 & 0 & 0 & 0 & 70 & 0 & 0 & 0 & 0 & 0 \\ & 156 & 0 & 0 & 0 & 22l & 0 & 54 & 0 & 0 & 0 & -13l \\ & & 156 & 0 & -22l & 0 & 0 & 0 & 54 & 0 & 13l & 0 \\ & & & 140 \frac{J}{A} & 0 & 0 & 0 & 0 & 0 & 70 \frac{J}{A} & 0 & 0 \\ & & & & 4l^2 & 0 & 0 & 0 & -13l & 0 & -3l^2 & 0 \\ & & & & & 4l^2 & 0 & 13l & 0 & 0 & 0 & -3l^2 \\ & & & & & & 140 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & & 156 & 0 & 0 & 0 & -22l \\ & & & & & & & & 156 & 0 & 22l & 0 \\ & & & & & & & & & 140 \frac{J}{A} & 0 & 0 \\ & & & & & & & & & & 4l^2 & 0 \\ & & & & & & & & & & & 4l^2 \end{bmatrix}$$

And the geometric stiffness is

$$K_g = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & \frac{6F_x}{5l} & 0 & \frac{M_{y1}}{l} & 0 & \frac{F_x}{10} & 0 & -\frac{6F_x}{5l} & 0 & \frac{M_{y2}}{l} & 0 & \frac{F_x}{10} \\ & & \frac{6F_x}{5l} & \frac{M_{z1}}{l} & -\frac{F_x}{10} & 0 & 0 & 0 & -\frac{6F_x}{5l} & \frac{M_{z2}}{l} & -\frac{F_x}{10} & 0 \\ & & & \frac{F_x I_{xx}}{Al} & -\frac{V_y l}{6} & -\frac{V_z l}{6} & 0 & -\frac{M_{y1}}{l} & -\frac{M_{z1}}{l} & -\frac{F_x I_{xx}}{Al} & \frac{V_y l}{6} & \frac{V_z l}{6} \\ & & & & \frac{2F_x l}{15} & 0 & 0 & 0 & \frac{F_x}{10} & \frac{V_y l}{6} & -\frac{F_x l}{30} & 0 \\ & & & & & \frac{2F_x l}{15} & 0 & -\frac{F_x}{10} & 0 & \frac{V_z l}{6} & 0 & -\frac{F_x l}{30} \\ & & & & & & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & & \frac{6F_x}{5l} & 0 & -\frac{M_{y2}}{l} & 0 & -\frac{F_x}{10} \\ & & & & & & & & \frac{6F_x}{5l} & -\frac{M_{z2}}{l} & -\frac{F_x}{10} & 0 \\ & & & & & & & & & \frac{F_x I_{xx}}{Al} & -\frac{V_y l}{6} & -\frac{V_z l}{6} \\ & & & & & & & & & & \frac{2F_x l}{15} & 0 \\ & & & & & & & & & & & \frac{2F_x l}{15} \end{bmatrix}$$

where

$$I_{xx} = I_{yy} + I_{zz}$$

Non-symmetric Beam Sections

In a beam with a symmetric section the bending properties depend only on the I_{yy} and I_{zz} terms. If such a beam is loaded in the 'y' or 'z' axis the deflection is in the direction of the loading.

When the section is not symmetric and is loaded in the 'y' or 'z' direction there is a component of deflection orthogonal to the loading. This is because the bending properties depend on I_{yy}, I_{zz} and I_{yz} .

By rotating the section to principal axes this cross term can be omitted and if the beam is loaded in the 'u' or 'v' (principal) axis the deflection is in the direction of the loading. In this case the stiffness matrix for the element is calculated using the principal second moments of area and is then rotated into the element local axis system.

For a beam with a non-symmetric section the user must consider if the beam is restrained (so that deflections are constrained to be in the direction of the loading) or if it will act in isolation (resulting in deflections orthogonal to the loading).

If the beam is to act as constrained the user should use the 'Ignore lyz' option. In this case the I_{yy} and I_{zz} values are used and the I_{yz} value is discarded.

If the beam is to act in isolation the user should not use the 'Ignore lyz' option. In this case the stiffness matrix for the element is calculated using the principal second moments of area and is then rotated into the element local axis system. In this case GSA assumes that the k_u value is the same as the k_y or k_z - whichever of 'y' or 'z' is closest to the 'u' axis, and likewise for the k_v value.

Where the user has specified section modifiers on the I or the k values there is no way to transfer these modifiers to the 'u' and 'v' directions. In these situations 'Ignore lyz' must be used.

All catalogue and standard sections except angles are symmetric. Explicit sections are assumed to be defined such that the principal and local axes coincide so there is no I_{yz} . Geometric (perimeter and line segment) sections are assumed to be non-symmetric.

Spring Elements

Spring element can one of several types.

The general spring has 6 degrees of freedom. For a linear spring the general form of the stiffness matrix is

$$\mathbf{K} = \begin{bmatrix} k_x & 0 & 0 & 0 & 0 & 0 & -k_x & 0 & 0 & 0 & 0 & 0 \\ & k_y & 0 & 0 & 0 & 0 & 0 & -k_y & 0 & 0 & 0 & 0 \\ & & k_z & 0 & 0 & 0 & 0 & 0 & -k_z & 0 & 0 & 0 \\ & & & k_{xx} & 0 & 0 & 0 & 0 & 0 & -k_{xx} & 0 & 0 \\ & & & & k_{yy} & 0 & 0 & 0 & 0 & 0 & -k_{yy} & 0 \\ & & & & & k_{zz} & 0 & 0 & 0 & 0 & 0 & -k_{zz} \\ & & & & & & k_x & 0 & 0 & 0 & 0 & 0 \\ & & & & & & & k_y & 0 & 0 & 0 & 0 \\ & & & & & & & & k_z & 0 & 0 & 0 \\ & & & & & & & & & k_{xx} & 0 & 0 \\ & & & & & & & & & & k_{yy} & 0 \\ & & & & & & & & & & & k_{zz} \end{bmatrix}$$

If non-linear spring are required these can be generated by assigning a material curve (force/displacement or moment/rotation) to each of the degrees of freedom.

Axial / torsional spring

For an axial or torsional spring only the k_x or k_{xx} terms are specified, the rest are assumed to be zero.

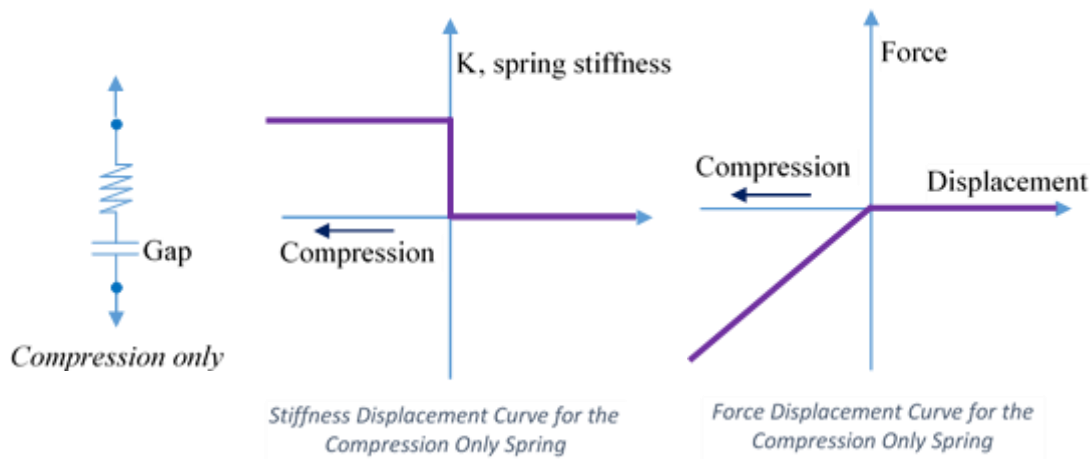
Spring matrix

For a ground spring a full set of stiffness terms can be defined for the upper triangle. This allows interaction between axial and bending terms. The stiffness matrix is then of the form

$$\mathbf{K} = \begin{bmatrix} k_{x,x} & k_{x,y} & k_{x,z} & k_{x,xx} & k_{x,yy} & k_{x,zz} \\ & k_{y,y} & k_{y,z} & k_{y,xx} & k_{y,yy} & k_{y,zz} \\ & & k_{z,z} & k_{z,xx} & k_{z,yy} & k_{z,zz} \\ & & & k_{xx,xx} & k_{xx,yy} & k_{xx,zz} \\ & & & & k_{yy,yy} & k_{yy,zz} \\ & & & & & k_{zz,zz} \end{bmatrix}$$

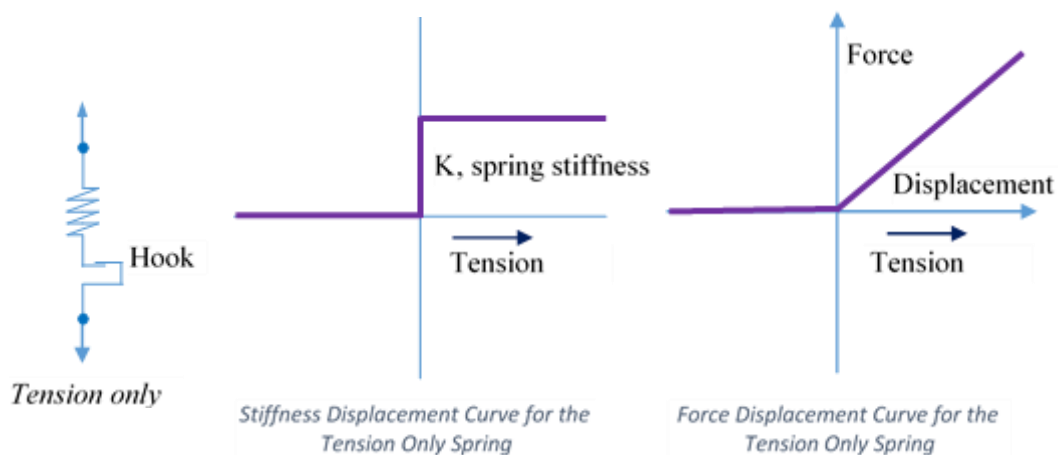
Compression-only spring

Compression only spring elements are uniaxial nonlinear spring elements. These elements have no stiffness when subjected tensile loading/displacements. Under tensile loading these elements will create a gap.



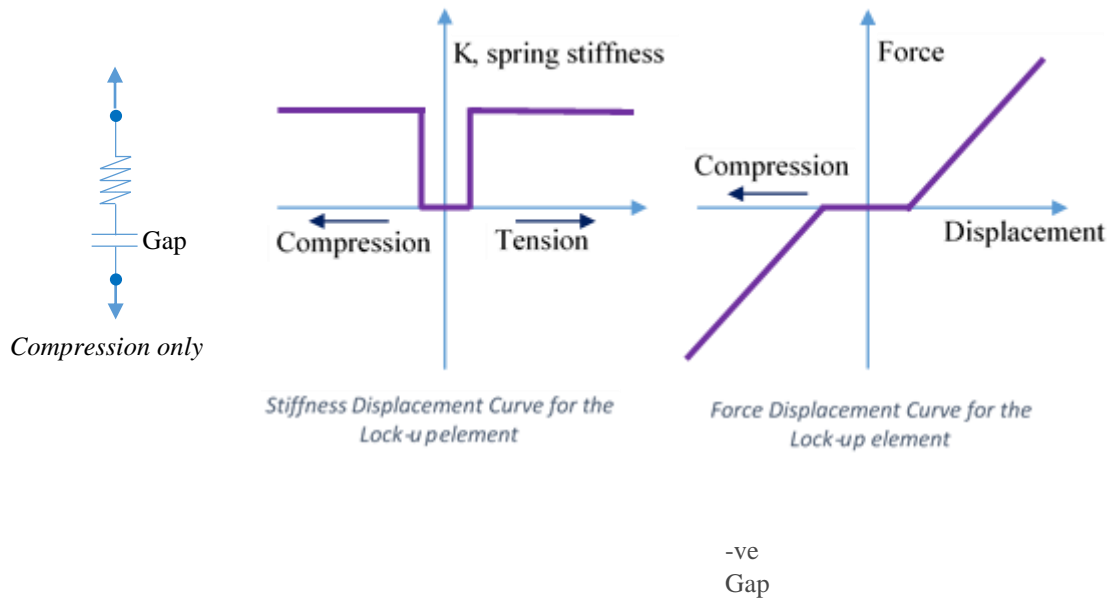
Tension-only spring

Tension only spring elements are also uniaxial nonlinear spring elements. These elements have no stiffness under compressive loading.



Lock-up

Lock-up elements are also uniaxial spring elements with specified initial gaps. Lock-up elements can be visualized as the combination of compression only and tension only spring with initial gaps. These spring elements do not offer any stiffness while the displacements are within the specified gap. Positive Gap will offer no resistance in the tension side and vice versa. Both positive and negative gaps have to be specified as positive values in GSA.

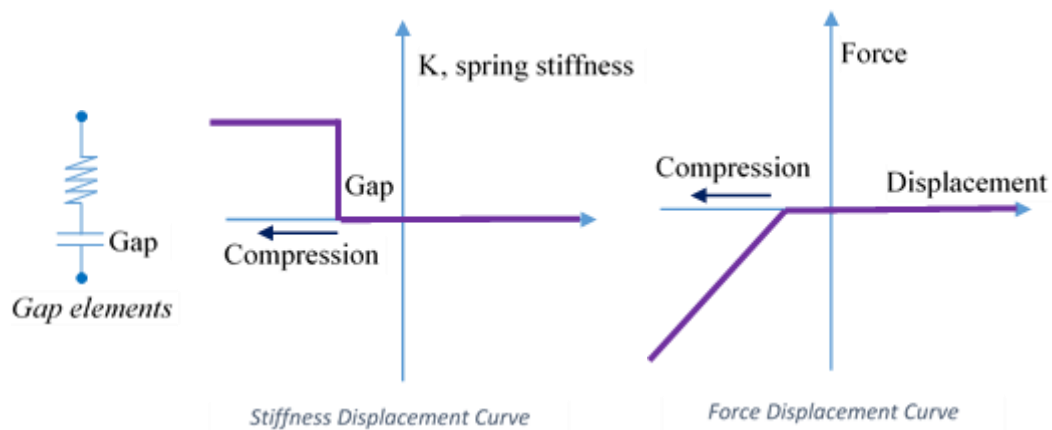


Connector

Connector elements are linear elastic spring elements. These elements connect a pair of nodes in all translations as well as rotations. The size of the stiffness matrix is 12x12 and the same will be automatically calculated based on the stiffness of the surrounding elements. The resulting stiffness is large enough to produce negligible displacements and at the same time it will avoid the numerical instabilities.

Gap elements

Gap elements are the compression only spring elements with initial gap. These elements will have zero stiffness in the analysis while the displacements are within the specified gap. The stiffness in other directions (local y and z directions) are ignored. These elements are useful in contact problems if frictional sliding is not a concern.

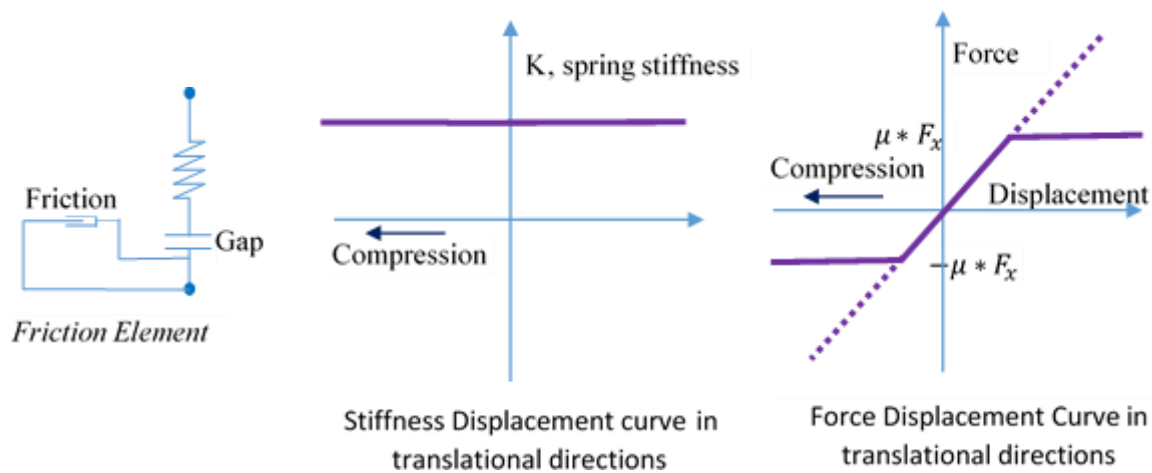


Friction elements

Friction elements are three direction translational spring elements consisting of one axial and two translational directions. In axial direction, these elements works as Gap elements. In translational direction these elements behaves as elastic plastic elements. Plastic limit is calculated using coulomb friction law

$$f_{y,z} = k \times \Delta u_{y,z} \leq \mu \times f_x$$

Where f_x is the force in axial direction as calculated for the gap element.



Springs are assumed to be massless.

Damper Elements

Damper element are elements with no stiffness but with viscosity, so these do not form stiffness matrices but instead a damping matrix. The general form of the damping matrix is

$$\mathbf{C} = \begin{bmatrix} c_x & 0 & 0 & 0 & 0 & 0 & -c_x & 0 & 0 & 0 & 0 & 0 \\ & c_y & 0 & 0 & 0 & 0 & 0 & -c_y & 0 & 0 & 0 & 0 \\ & & c_z & 0 & 0 & 0 & 0 & 0 & -c_z & 0 & 0 & 0 \\ & & & c_{xx} & 0 & 0 & 0 & 0 & 0 & -c_{xx} & 0 & 0 \\ & & & & c_{yy} & 0 & 0 & 0 & 0 & 0 & -c_{yy} & 0 \\ & & & & & c_{zz} & 0 & 0 & 0 & 0 & 0 & -c_{zz} \\ & & & & & & c_x & 0 & 0 & 0 & 0 & 0 \\ & & & & & & & c_y & 0 & 0 & 0 & 0 \\ & & & & & & & & c_z & 0 & 0 & 0 \\ & & & & & & & & & c_{xx} & 0 & 0 \\ & & & & & & & & & & c_{yy} & 0 \\ & & & & & & & & & & & c_{zz} \end{bmatrix}$$

For an axial or torsional damper only the c_x or c_{xx} terms are specified, the rest are assumed to be zero.

Dampers are assumed to be massless.

Mass Elements

Mass elements can have only mass and inertia. In the normal case the mass matrix is

$$\mathbf{M} = \begin{bmatrix} m & 0 & 0 \\ & m & 0 \\ & & m \end{bmatrix}$$

but if mass modifiers are included the mass matrix becomes

$$\mathbf{M} = \begin{bmatrix} m_x & 0 & 0 \\ & m_y & 0 \\ & & m_z \end{bmatrix}$$

Inertia is a tensor quantity with the following terms

$$\mathbf{I} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ & I_{yy} & I_{yz} \\ & & I_{zz} \end{bmatrix}$$

where the terms in the inertia tensor are defined as

$$\begin{aligned} I_{xx} &= \int_V \rho [(y - y_c)^2 + (z - z_c)^2] dV & I_{xy} &= I_{yx} = \int_V \rho (x - x_c)(y - y_c) dV \\ I_{yy} &= \int_V \rho [(z - z_c)^2 + (x - x_c)^2] dV & I_{yz} &= I_{zy} = \int_V \rho (y - y_c)(z - z_c) dV \\ I_{zz} &= \int_V \rho [(x - x_c)^2 + (y - y_c)^2] dV & I_{zx} &= I_{xz} = \int_V \rho (z - z_c)(x - x_c) dV \end{aligned}$$

where (x_a, y_a, z_a) are the coordinates of the centre of mass. If non-zero values are specified for the off-diagonal terms, it is important that these are consistent with the diagonal terms. If this is not done the principal inertia values can become negative. The inertia matrix is never modified for directions.

2D Elements

An irregularly shaped continuum defined by a boundary and areas of loading has to be subdivided into a mesh of finite elements. The size and shape of the element is chosen so that the approximate stiffness implied by the finite element is close enough to the actual stiffness of the continuum in that region. Thus the finite element size and shape will be determined by the

type of element being used, the shape of the boundary if in a region close to the boundary, the loading applied and the experience of the user.

Element stiffness

To generate a stiffness matrix for a curvilinear quadrilateral or triangular element a new approach must be used. Most finite element codes used an approach based on isoparametric or similar elements.

In an isoparametric element the element displacements are interpolated in the same way as the geometry, eg a plane stress element. In a superparametric or degenerate isoparametric element the interpolation on the geometry is of a higher order than the interpolation of the displacements, eg a plate element. In a subparametric element the interpolation of the geometry is of a lower order than the interpolation of the displacements, e.g. an eight noded straight sided quadrilateral element, where a different geometric interpolation function is used for the geometry from that for the displacements. The term isoparametric is often used as a general term to cover all these element types.

For a plane stress problem, we can establish a material matrix C that relates stress and strain. The displacements in a local coordinate system are

$$\mathbf{u} = \{u, v\}$$

the strains are

$$\boldsymbol{\varepsilon} = \{\boldsymbol{\varepsilon}_{xx}, \boldsymbol{\varepsilon}_{yy}, \boldsymbol{\gamma}_{xy}\}$$

and the stresses are

$$\boldsymbol{\sigma} = \{\boldsymbol{\sigma}_{xx}, \boldsymbol{\sigma}_{yy}, \boldsymbol{\tau}_{xy}\}$$

For an elastic-isotropic material the material matrix is

$$\mathbf{C}_p = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & & \\ \nu & 1 & & \\ & & \frac{1-\nu}{2} & \\ & & & \frac{1-\nu}{2} \end{bmatrix}$$

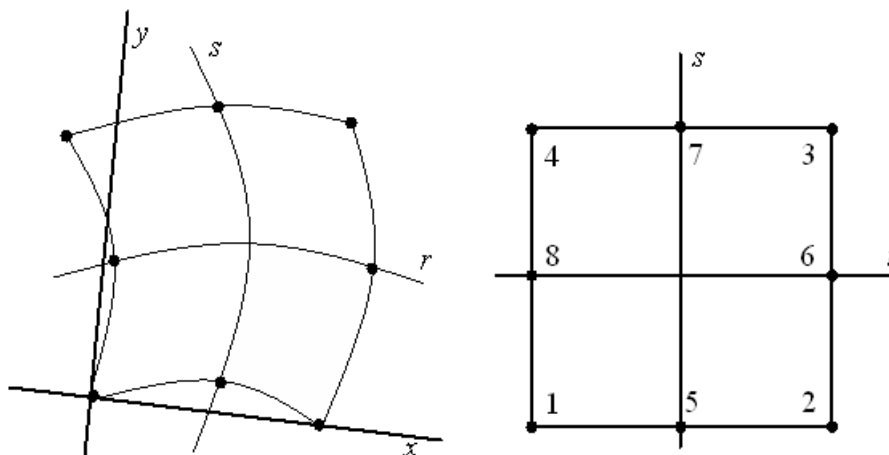
where E, ν are the Young's modulus and Poisson's ratio respectively.

Note that there is an out of plane strain $\boldsymbol{\varepsilon}_{zz}$, which we can ignore as it plays no part in the element formulation.

The strains are defined in terms of the displacements as

$$\boldsymbol{\varepsilon}_{xx} = \frac{\partial u}{\partial x} \quad \boldsymbol{\varepsilon}_{yy} = \frac{\partial v}{\partial y} \quad \boldsymbol{\gamma}_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}$$

The simplest elements to consider are the 4 noded and 8 noded quadrilateral elements, of which the 4 noded element can be considered a simplification of the 8 noded element. A typical 8 noded element is shown below. The element has an arbitrary local coordinate system based on the location of the nodes and the element property axis (x, y), and a natural (curvilinear) coordinate system (r, s) based on the topology of the nodes.



We can set up interpolation functions to interpolate the geometry as follows

$$x_s(r, s) = \sum_i h_i x_{m,i}$$

where the h_i are the interpolation functions defined below.

These interpolation functions are chosen so that

$$h_i = \begin{cases} 1 \\ 0 \end{cases} \text{ when node is } \begin{cases} i \\ \neq i \end{cases}$$

As the elements are isoparametric we use the same interpolation function for the displacements so the displacement in the element is related to the nodal displacements by

$$u_s(r, s) = \sum_i h_i u_{m,i}$$

To evaluate the stiffness matrix we need the strain-displacement transformation matrix. The element displacements are obtained in terms of derivatives of the element displacements with respect to the local coordinate system (x, y) . Because the elements displacements are in the natural coordinate system (r, s) we need to relate the derivatives in the local coordinate system to those in the natural coordinate system. We can write an equation for the derivative with respect to x in terms of derivatives with respect to (r, s)

$$\frac{\partial}{\partial x} = \frac{\partial}{\partial r} \frac{\partial r}{\partial x} + \frac{\partial}{\partial s} \frac{\partial s}{\partial x}$$

to establish these derivatives we use the chain rule to set up the following relationship

$$\begin{bmatrix} \frac{\partial}{\partial r} \\ \frac{\partial}{\partial s} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} \\ \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix}$$

or in matrix notation

$$\frac{\partial}{\partial \mathbf{r}} = \mathbf{J} \frac{\partial}{\partial \mathbf{x}}$$

where \mathbf{J} is the Jacobian operator relating natural coordinate derivatives to local coordinate derivatives. Given that we know x and y in terms of the interpolation functions the Jacobian operator is easily found

$$\frac{\partial}{\partial \mathbf{x}} = \mathbf{J}^{-1} \frac{\partial}{\partial \mathbf{r}}$$

This requires that the inverse of the Jacobian exist, which requires that there is a one to one correspondence between natural and local coordinates. This will be the case provided the element is not grossly distorted from a square and that it does not fold back on itself.

We can then evaluate

$$\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}, \frac{\partial v}{\partial x}, \frac{\partial v}{\partial y}$$

and thus we can construct the strain-displacement transformation matrix, \mathbf{B}

$$\boldsymbol{\varepsilon} = \mathbf{B}\mathbf{u}$$

where \mathbf{u} is the vector of nodal displacements. The element stiffness corresponding to the local element degrees of freedom is then

$$\mathbf{K} = \int_V \mathbf{B}^T \mathbf{C} \mathbf{B} dV$$

The elements of \mathbf{B} are functions of the natural coordinate system, (r, s) . Therefore the volume integration extends over the natural coordinate volume so the volume differential needs to be written in terms of the natural coordinates

$$dV = \det \mathbf{J} dr ds$$

The volume integral is not normally amenable to an explicit integration so normally a numerical integration technique is used. The integral can be written

$$\mathbf{K} = \iint_{r,s} \mathbf{F} dr ds$$

where

$$\mathbf{F} = \mathbf{B}^T \mathbf{C} \mathbf{B} \det \mathbf{J}$$

and the integral is performed in the natural coordinate system of the element. This is convenient as the limits of the integration are then ± 1 . The stiffness can then be calculated

$$\mathbf{K} = \sum \alpha_{ij} \mathbf{F}_{ij}$$

where the matrix \mathbf{F} is evaluated at the Gaussian integration points (r_i, s_i) and α_{ij} are Gaussian weights.

In a similar way the mass matrix and the load vectors are established.

$$\mathbf{M} = \int_V \rho \mathbf{H}^T \mathbf{H} dV$$

$$\mathbf{r}_b = \int_V \mathbf{H}^T \mathbf{f}_b dV \quad \mathbf{r}_s = \int_S \mathbf{H}_s^T \mathbf{f}_s dS \quad \mathbf{r}_i = \int_V \mathbf{B}^T \boldsymbol{\tau}_i dV$$

where \mathbf{H} is a matrix of interpolation functions and $\mathbf{f}_b, \mathbf{f}_s, \boldsymbol{\tau}_i$ are the body force vector, surface force vector and initial stress vector respectively.

Geometric stiffness matrix of shell element

The geometric stiffness matrix is calculated from:

$$\begin{aligned} \mathbf{K}_g &= \int_A \mathbf{G}^T \mathbf{N} \mathbf{G} dA \\ &= \iint_{r,s} \mathbf{G}^T \mathbf{N} \mathbf{G} \det \mathbf{J} dr ds \end{aligned}$$

where

$$\mathbf{N} = \begin{bmatrix} N_x & N_{xy} \\ N_{yx} & N_y \end{bmatrix}$$

With (N_x, N_y, N_{xy}) are the in-plane forces of the shell element in x, y and xy shear directions respectively.

$$\mathbf{G} = [G_1 \quad G_2 \quad \dots \quad G_n]$$

and n is the number of nodes of the elements

$$\mathbf{G}_i = \begin{bmatrix} 0 & 0 & \frac{\partial h_i}{\partial x} & 0 & 0 & 0 \\ 0 & 0 & \frac{\partial h_i}{\partial y} & 0 & 0 & 0 \end{bmatrix}$$

2D Plate Elements

The formulation of 2D plate and shell elements considers both in-plane and transverse (out-of-plane) displacements. Following Mindlin-Reissner plate theory, in addition to the bending strains we consider the effect of transverse shear strain in our complete expression for the element strain

$$\begin{Bmatrix} \gamma_{xz} \\ \gamma_{yz} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial w}{\partial x} - \beta_x \\ \frac{\partial w}{\partial y} - \beta_y \end{Bmatrix}$$

where w is the out-of-plane displacement and β is introduced as an independent variable to express the section rotations (i.e. rotations of the transverse normals about the local x and y axes).

We can define separate material matrices \mathbf{C}_b and \mathbf{C}_s that relate stress and strain for the pure bending and shear strains respectively and so the pure bending moments and shear forces can be written

$$\begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \tau_{xy} \end{Bmatrix} = -z \mathbf{C}_b \begin{Bmatrix} \frac{\partial \beta_x}{\partial x} \\ \frac{\partial \beta_y}{\partial y} \\ \frac{\partial \beta_x}{\partial y} + \frac{\partial \beta_y}{\partial x} \end{Bmatrix}$$

and

$$\begin{Bmatrix} \tau_{xz} \\ \tau_{yz} \end{Bmatrix} = \mathbf{C}_s \begin{Bmatrix} \frac{\partial w}{\partial x} - \beta_x \\ \frac{\partial w}{\partial y} - \beta_y \end{Bmatrix}$$

respectively.

In this way we can then express the local element stiffness matrix as a summation of the in-plane and out-of-plane stiffness's

$$\mathbf{K} = \int_V (\mathbf{B}_\kappa^T \mathbf{C}_b \mathbf{B}_\kappa + \mathbf{B}_\gamma^T \mathbf{C}_s \mathbf{B}_\gamma) dV$$

where \mathbf{B}_κ and \mathbf{B}_γ represent the strain-displacement transforms for the bending and shear components respectively.

While brief, this outlines the basic approach to the Mindlin-Reissner 2D element stiffness formulation. In GSA we label this concisely as the Mindlin formulation.

MITC Element Formulation

We find that the Mindlin formulation is an effective approach for 2D parabolic elements where the 8-noded element accommodates sufficient terms in the stiffness matrix to sufficiently define the behaviour of the element numerically. However the same formulation defined over a linear element becomes noticeably more problematic where the absence of available terms in our 4-noded stiffness matrix leads to numerical difficulties in expressing the same element behaviour. Specifically we find a difficulty in attempting to represent the transverse shear strain terms. Analytically as before we can express the transverse shear strain as

$$\begin{Bmatrix} \gamma_{xz} \\ \gamma_{yz} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial w}{\partial x} - \beta_x \\ \frac{\partial w}{\partial y} - \beta_y \end{Bmatrix}$$

although numerically, the difference in the order of terms for the shear strain may lead to artificial stiffening of the element where the shear terms are numerically constrained from approaching zero. See reference 1 in the bibliography for further information. This restriction would be particularly noticeable where the thickness of the plate is small.

A widely practiced remedy is to under-integrate the shear term and while effective, its use is at the cost of reduced accuracy and stability for the element. The problem of stability alone is often of greatest concern where the phenomenon of hourglassing can become apparent in elements where the thickness to length ratio is large.

An alternative formulation was put forward by Bathe and Dvorkin and has been found to be especially effective at resolving these difficulties. The formulation is extendable to higher order elements although we find the approach is most effective when resolving the difficulties most apparent in linear elements. The formulation is based upon the theory of Mixed Interpolated

Tensorial Components (MITC). For the pure displacement-based Mindlin formulation we use the independent variables

$$w = \sum_{i=1}^n \mathbf{h}_i^T \mathbf{w}_i$$

$$\beta = \sum_{i=1}^n \mathbf{h}_i^T \theta_i$$

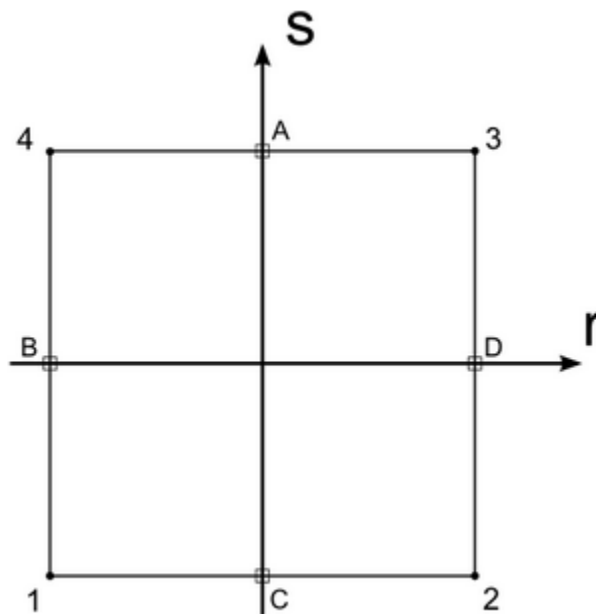
where Bathe now introduces a separate independent variable to represent the transverse shear term

$$\gamma = \sum_{i=1}^n \mathbf{h}_i^{*T} \gamma_{Pi}$$

We use \mathbf{h}_i^* to represent an additional set of interpolation functions for our new variable γ which we find by a direct evaluation of the shear strain at points P_i , that is

$$\gamma_{Pi} = \left(\frac{\partial w}{\partial x} - \beta \right)_{Pi}$$

For a linear 2D element we obtain direct values for the shear strain at four points A, B, C, D on the element and so we evaluate the displacement and section rotations at these points through direct interpolation.



We can then construct the interpolations

$$\gamma_{rz} = \frac{1}{2}(1+s)\gamma_{rz}^A + \frac{1}{2}(1-s)\gamma_{rz}^C$$

$$\gamma_{sz} = \frac{1}{2}(1+r)\gamma_{sz}^D + \frac{1}{2}(1-r)\gamma_{sz}^B$$

using direct values for the shear strains obtained at the four points. This replaces our original expression for the shear terms and we continue to construct the local stiffness matrix as normal in a similar approach as before in 2D isoparametric elements. It is lastly worthwhile to note that the interpolation above assumes our element is in the isoparametric coordinate system. Further transformations are necessary to interpolate the shear strains directly through an arbitrary element in local space. Indeed, when this is done the element shows considerably improved predicative capabilities for distorted elements.

Element shape functions

The element shape functions for 2D elements are

$$u_s(r, s) = \sum_i h_i u_{m,i}$$

For a tri-3 element these are

$$h_1 = 1 - r - s$$

$$h_2 = r$$

$$h_3 = s$$

For a quad-4 element these are

$$h_1 = \frac{1}{4}(1-r)(1-s)$$

$$h_2 = \frac{1}{4}(1+r)(1-s)$$

$$h_3 = \frac{1}{4}(1+r)(1+s)$$

$$h_4 = \frac{1}{4}(1-r)(1+s)$$

For a tri-6 element these are

$$h_1 = 1 - r - s - \frac{1}{2}h_4 - \frac{1}{2}h_6$$

$$h_2 = r - \frac{1}{2}h_4 - \frac{1}{2}h_5$$

$$h_3 = s - \frac{1}{2}h_5 - \frac{1}{2}h_6$$

$$h_4 = 4r(1-r-s)$$

$$h_5 = 4rs$$

$$h_6 = 4s(1-r-s)$$

For a quad-8 element these are

$$\begin{aligned}
h_1 &= \frac{1}{4}(1-r)(1-s) - \frac{1}{2}h_5 - \frac{1}{2}h_8 \\
h_2 &= \frac{1}{4}(1+r)(1-s) - \frac{1}{2}h_5 - \frac{1}{2}h_6 \\
h_3 &= \frac{1}{4}(1+r)(1+s) - \frac{1}{2}h_6 - \frac{1}{2}h_7 \\
h_4 &= \frac{1}{4}(1-r)(1+s) - \frac{1}{2}h_7 - \frac{1}{2}h_8 \\
h_5 &= \frac{1}{4}(1-r^2)(1-s) \\
h_6 &= \frac{1}{4}(1+r)(1-s^2) \\
h_7 &= \frac{1}{4}(1-r^2)(1+s) \\
h_8 &= \frac{1}{4}(1-r)(1-s^2)
\end{aligned}$$

For a brick-8 element these are

$$\begin{aligned}
h_1 &= \frac{1}{4}(1-r)(1-s)(1-t) \\
h_2 &= \frac{1}{4}(1+r)(1-s)(1-t) \\
h_3 &= \frac{1}{4}(1+r)(1+s)(1-t) \\
h_4 &= \frac{1}{4}(1-r)(1+s)(1-t) \\
h_5 &= \frac{1}{4}(1-r)(1-s)(1+t) \\
h_6 &= \frac{1}{4}(1+r)(1-s)(1+t) \\
h_7 &= \frac{1}{4}(1+r)(1+s)(1+t) \\
h_8 &= \frac{1}{4}(1-r)(1+s)(1+t)
\end{aligned}$$

Wall Element

Wall element is a super element, internally it is meshed by 4x4 quad8 shell elements, and therefore the size of wall elements can be relatively large without losing the accuracy of analysis results. Typically a wall from one floor to the next can be modelled by a single wall element rather than a shell element mesh. The internal nodes of wall elements are condensed out and do not contribute to the number of degree of freedom in global analysis, but the middle nodes along the edges are considered in global analysis, but they are hidden for users, the middle nodes along edges do contribute to the total number of degree of freedom of the model.

As wall elements are large, the constraints at the hidden nodes along the edges will affect wall element stiffness and analysis results. The constraints of the middle nodes along the edges are considered in the following way, if the two corner nodes of an edge have the same nodal constraints, e.g. both nodes have x and zz constraints, then all the middle nodes along this edge will be constrained in x and zz. However if only one of the two nodes of the edge has constraint in a certain direction, the middle nodes along this edge will be free, e.g. if one corner node has x constraint and the other corner node has zz constraint, then all the middle nodes along this edge will be free.

The above constraint principle also applies to rigid constraints, link elements and joints, i.e. if both corner nodes of an edge are in the same rigid constraint, the same link or the same joint, all the hidden nodes along the edge will be also in the same rigid constraint, the same link or the

same joint respectively. This will guarantee there is no incompatibility between the displacements along the edge and the two edge node.

2D Element Shape Checks

A number of element checks are carried out by GSA prior to a GSS analysis. Other analysis programs may have different limits but the same principles apply. For GSS the following warnings, severe warnings and errors are produced.

Triangle

Warning

$$5 < r_{\max} \leq 15$$

$$15 \leq \theta_{\min} < 30$$

$$150 \leq \theta_{\max} \leq 165$$

Severe warning

$$r_{\max} > 15$$

$$\theta_{\min} < 15$$

Error

$$\theta_{\max} > 165$$

Quad

Warning

$$5 < r_{\max} \leq 15$$

$$25 \leq \theta_{\min} < 45$$

$$135 \leq \theta_{\max} \leq 155$$

$$0.00001 < h_{\max} \leq 0.01$$

Severe warning

$$r_{\max} > 15$$

$$\theta_{\min} < 25$$

$$h_{\max} > 0.01$$

Error

$$\theta_{\max} > 155$$

where

r_{\max} longest side / shortest side

θ_{\min} minimum angle

θ_{\max} maximum angle

h_{\max} distance out of the plane of the element of edge 1 / longest side

Notes:

The distance out of plane of edge 1 is calculated as max max

$$h_{\max} = \frac{\mathbf{n} \cdot (\mathbf{c}_2 - \mathbf{c}_1)}{s_{\max}}$$

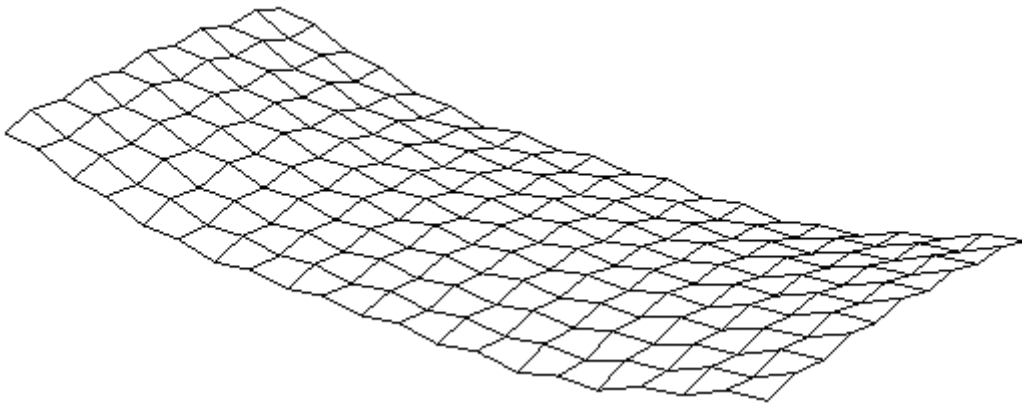
Where \mathbf{n} is the element normal, $\mathbf{c}_1, \mathbf{c}_2$ are the coordinates of the first and second corner nodes and s_{\max} is the length of the longest side of the element.

Mid-side node locations not checked but should be approximately halfway along edge.

No check on ratio thickness/shortest side.

Hourglassing

When Quad 4 elements with the Mindlin formulation are used in bending it is possible to encounter hourglassing problems. This is a problem that arises with under-integrated elements where there are insufficient stiffness terms to fully represent the stiffness of the element. The problem is noticeable in the results by an hourglass pattern in the mesh as shown below.



This problem is avoided using the MITC formulation for Quad 4 elements. This formulation uses a separate interpolation method for the transverse shear strains and provides considerably greater stability than the original Mindlin method. The original Mindlin method is kept in GSA for compatibility with previous models although for new models the MITC formulation is recommended.

Alternatively when parabolic accuracy is required Quad 8 elements are recommended. These also formulate elements that are still stiff in all modes of deformation, even when under-integrated.

Link Element

Link elements are different to the other element types in that they apply a constraint between a pair of nodes. The master node is the first node specified in the element topology and this is the node that is retained in the solution. The other node (the slave node) is related back to the master node.

$$\mathbf{u}_s = \mathbf{T} \mathbf{u}_m$$

The degrees of freedom at the slaves that are linked will depend on the type of link. The link allows the slave node to be fixed (where the rotations at the slave node depend on the rotation of the master) or pinned (where the rotations at the slave node are independent of the rotation of the master). The master node is always has the rotations linked to the rest of the structure. The links can act in all directions or be restricted to act in a plane (xy, yz or zx) where the nodes are rigidly connected for motion in the plane but are independent for out of plane motion.

The constraint conditions are summarised below:

All directions

$$\begin{bmatrix} 1 & 0 & 0 & 0 & z & -y \\ 0 & 1 & 0 & -z & 0 & x \\ 0 & 0 & 1 & y & -x & 0 \\ 0 & 0 & 0 & \delta & 0 & 0 \\ 0 & 0 & 0 & 0 & \delta & 0 \\ 0 & 0 & 0 & 0 & 0 & \delta \end{bmatrix} \quad \text{where } \delta = 1/0 \text{ if fixed / pinned}$$

Plane (xy plane as example)

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & -y \\ 0 & 1 & 0 & 0 & 0 & x \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \delta \end{bmatrix} \quad \text{where } \delta = 1/0 \text{ if fixed / pinned}$$

Loads applied to link elements will be correctly transferred to the master degree of freedom as a force + moment so no spurious moments result.

The inertia properties of a link element can be calculated from the masses at the nodes attached to the rigid element as follows. The mass is

$$m = \sum_i m_i$$

and the coordinates of the centre of mass are

$$x_c = \frac{\sum_i m_i x_i}{m} \quad y_c = \frac{\sum_i m_i y_i}{m} \quad z_c = \frac{\sum_i m_i z_i}{m}$$

The inertia about the global origin is then

$$\begin{aligned} \bar{I}_{xx} &= \sum_i m_i (y_i^2 + z_i^2) & \bar{I}_{xy} &= -\sum_i m_i x_i y_i \\ \bar{I}_{yy} &= \sum_i m_i (z_i^2 + x_i^2) & \bar{I}_{yz} &= -\sum_i m_i y_i z_i \\ \bar{I}_{zz} &= \sum_i m_i (x_i^2 + y_i^2) & \bar{I}_{zx} &= -\sum_i m_i z_i x_i \end{aligned}$$

and relative to the centre of mass (x_c, y_c, z_c) this is

$$\begin{aligned}
 I_{xx} &= \sum_i m_i \left((y_i - y_c)^2 + (z_i - z_c)^2 \right) & \bar{I}_{xy} &= -\sum_i m_i (x_i - x_c)(y_i - y_c) \\
 I_{yy} &= \sum_i m_i \left((z_i - z_c)^2 + (x_i - x_c)^2 \right) & \bar{I}_{yz} &= -\sum_i m_i (y_i - y_c)(z_i - z_c) \\
 I_{zz} &= \sum_i m_i \left((x_i - x_c)^2 + (y_i - y_c)^2 \right) & \bar{I}_{zx} &= -\sum_i m_i (z_i - z_c)(x_i - x_c)
 \end{aligned}$$

The constraint equations for a link element assume small displacements. When large displacements are applied to a link element the constraint equations no longer apply and the links between slave and master get stretched. This effect can be noticeable in a dynamic analysis where the results are scaled to an artificially large value. When these are scaled to realistic value this error should be insignificant.

Element Mass

The mass matrix for an element can be derived as described above. This is known as a consistent mass matrix. In many situations it is convenient to simplify the mass matrix. One way of doing this is to diagonalize the mass matrix. In this case all the terms relating to rotations are zeroed and then a row summation is carried out on the remaining entries.

$$m_{ii} = \sum_j m_{ij}$$

This has the effect of lumping all the mass at the nodes. The other simplification that is used is to ignore the mass of all elements except lumped masses.

With both these simplifications it may be possible to diagonalize the structure mass matrix, however this is not possible if the inertias of lumped mass elements are non-zero or if there are rigid elements in the structure. (Rigid elements generate off diagonal terms when the masses are replaced by inertias at the master.)

Element Stiffness

The element stiffness is created initially in the local axes of the element. This gives a square symmetric matrix.

$$\mathbf{f}_e = \mathbf{K}_e \mathbf{u}_e$$

This need to be transformed before it is added to the structure stiffness matrix, however some degrees of freedom are 'released' and these are retained on the element. This is represented by the matrix equation

$$\begin{Bmatrix} \mathbf{f}_s \\ \mathbf{f}_e \end{Bmatrix} = \begin{bmatrix} \mathbf{K}_{ss} & \mathbf{K}_{se} \\ \mathbf{K}_{es} & \mathbf{K}_{ee} \end{bmatrix} \begin{Bmatrix} \mathbf{u}_s \\ \mathbf{u}_e \end{Bmatrix}$$

The structure degrees of freedom need to be transformed to global directions, using the direction cosine array \mathbf{D} .

$$\begin{Bmatrix} \mathbf{f}_g \\ \mathbf{f}_e \end{Bmatrix} = \begin{bmatrix} \mathbf{D}^T \mathbf{K}_{ss} \mathbf{D} & \mathbf{D}^T \mathbf{K}_{se} \\ \mathbf{K}_{es} \mathbf{D} & \mathbf{K}_{ee} \end{bmatrix} \begin{Bmatrix} \mathbf{u}_g \\ \mathbf{u}_e \end{Bmatrix}$$

Offsets in global directions then relate the global structural degrees of freedom to the nodal degrees of freedom through a rigid transformation

$$\begin{Bmatrix} \mathbf{f}_n \\ \mathbf{f}_e \end{Bmatrix} = \begin{bmatrix} \mathbf{T}_T \mathbf{D}^T \mathbf{K}_{ss} \mathbf{D} \mathbf{T} & \mathbf{T}^T \mathbf{D}^T \mathbf{K}_{se} \\ \mathbf{K}_{es} \mathbf{D} \mathbf{T} & \mathbf{K}_{ee} \end{bmatrix} \begin{Bmatrix} \mathbf{u}_n \\ \mathbf{u}_e \end{Bmatrix}$$

Releases

Releases are applied at the nodes (or pseudo nodes in the case of offset elements) so that the elements do not have any moment connection. This can be applied to the elements by recognising that any moment applied to node is not resisted by the element. This condition is used to partition the element stiffness matrix.

$$\begin{Bmatrix} \mathbf{f}_s \\ \mathbf{f}_e \end{Bmatrix} = \begin{bmatrix} \mathbf{k}_{ss} & \mathbf{k}_{se} \\ \mathbf{k}_{es} & \mathbf{k}_{ee} \end{bmatrix} \begin{Bmatrix} \mathbf{u}_s \\ \mathbf{u}_e \end{Bmatrix}$$

Where the subscripts *s* refer to the structure and *e* refer to the element. Once partitioned the degrees of freedom related to the structure are combined into the structure stiffness matrix while the element degrees of freedom are included in the structure stiffness matrix (but do not interact with the stiffness matrix for any other element).

In the case of stiffnesses at the releases we add the stiffness terms for the release into the partitioned stiffness matrix. So in the case of a beam with releases at end 2 and stiffness associated with the released degrees of freedom the matrix is partitioned as below.

$$\begin{bmatrix} \mathbf{k}_{s11} & \mathbf{k}_{s12} \\ \mathbf{k}_{s21} & \mathbf{k}_{s22} \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{k}_{s11} & \tilde{\mathbf{k}}_{s12} & \tilde{\mathbf{k}}_{e12} \\ \tilde{\mathbf{k}}_{s21} & \tilde{\mathbf{k}}_{s22} & \\ \tilde{\mathbf{k}}_{e21} & & \tilde{\mathbf{k}}_{e22} \end{bmatrix} + \begin{bmatrix} & \mathbf{k}_r & -\mathbf{k}_r \\ -\mathbf{k}_r & & \mathbf{k}_r \end{bmatrix}$$

The stiffness terms at nodes where releases are applied are split, so

$$\mathbf{k}_{sij} \rightarrow \tilde{\mathbf{k}}_{sij} + \tilde{\mathbf{k}}_{eij}$$

The release stiffness terms are similar to a spring stiffness matrix:

$$\mathbf{k}_r = \begin{bmatrix} k_x & & & & & \\ & k_y & & & & \\ & & k_z & & & \\ & & & k_{xx} & & \\ & & & & k_{yy} & \\ & & & & & k_{zz} \end{bmatrix}$$

Offsets

Element offsets are defined as global vectors relative to the nodes. These locate pseudo nodes that define the flexible part of the element. Element stiffnesses are calculated for the flexible part and modified to give the required nodal stiffnesses. Release conditions are applied before the offsets (i.e. they are applied to the pseudo nodes). The element local axes are defined with respect to the flexible part of the element.

The offsets can then be considered as rigid links and a constraint equation set established linking pseudo nodes to actual nodes

$$\phi_{pseudo} = T \phi_{node}$$

where the transformation is based on the location of the pseudo node (x, y, z) relative to the actual node is

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 & z & -y \\ 0 & 1 & 0 & -z & 0 & x \\ 0 & 0 & 1 & y & -x & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The displacements at the pseudo nodes can then be related to the displacements at the actual nodes.

$$\mathbf{u}_{pseudo} = T \mathbf{u}_{node}$$

The forces at the nodes can be related to the forces at the pseudo nodes.

$$\mathbf{f}_{node} = T^T \mathbf{f}_{pseudo}$$

The element stiffness can then be modified on a node by node basis using

$$\mathbf{f}_{i,pseudo} = \mathbf{K}_{ij} \mathbf{u}_{j,pseudo}$$

$$\mathbf{f}_{i,node} = T_i^T \mathbf{K}_{ij} T_j \mathbf{u}_{j,node}$$

Spacer Elements & Sliding Cables

Spacer elements and cable elements are intended to be used as part of multi-noded super-elements or chains in the dynamic relaxation solver. To define the chain, or super-element, the program looks for spacer or cable elements with the same property number, and joins these at common nodes. So all the elements in a continuous length of spacer or cable must have a common property number, unique to that chain.

In other words each super-element is identified by a unique property number. A Super element cannot be discontinuous or bifurcated.

With spacer elements, the order for the input of elements is important as the “ratio” feature adjusts the relative distances between nodes and makes the first element the control: i.e. if the ratio is greater than one, the first element will be the smallest. If it is less than one, the first element will be the largest. The nodes for a spacer element should be defined so that they can be joined together to form a spacer in a head-to-tail sequence.

Generally spacers should not cross each other. If spacers are required in two directions, the spacers in the main direction should be carried through from one edge of a surface to the other and the spacers in the subsidiary direction should join nodes in adjacent main spacers.

Cables and Sliding

Cables are a hybrid form of cable element, intended for use in dynamic relaxation analysis of ‘real’ (not ‘form-finding’) structural models. In this analysis the cable is free to slide at internal nodes where cable element are connected. The cable, composed of a number of cable elements, is considered as a single element in dynamic-relaxation. The cable can freely slide around the internal nodes of the element as if the cable moves around a pulley. As a result, the tensile forces of all the cable elements in a cable are the same. If a clamp is needed along the length of a Sliding Cable, two Sliding Cable elements should be defined, joining at the clamped node.

The cable property is defined by a stiffness per unit length which is equal to EA where

E – elastic modulus

A – cable cross section area

The tensile force in the cable is calculated from

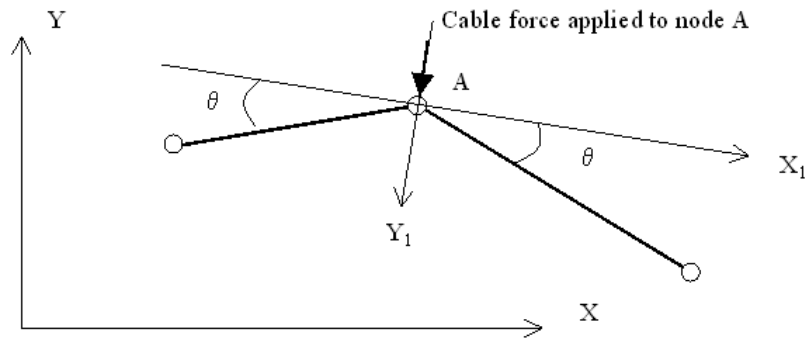
$$f = \begin{cases} EA \frac{l-l_0}{l_0} & l > l_0 \\ 0 & l \leq l_0 \end{cases}$$

in which

l_0 is the total unstressed length of the cable

l is the total deformed length of the cable

The nodal normal direction in a cable is defined by the line that is within the plane defined by the two legs and it evenly divides the angle composed by the two legs. Only the normal component of force is transferred between the nodes along a cable and the cable.



For the portion of cable shown above assume the cable elements are in the x-y plane.

$$F_A = T(x_1, y_1, z_1)$$

The cable force is constant along its length. Therefore the components of the cable elements along X_1 at node A are equal and opposite. And the resultant force in this direction is 0.

The component along Y_1 is applied to node A.

The component along Z_1 is 0 as the cable is in the x-y plane.

$$F_A = T(0, y_1, 0)$$

Free spacers and cables can be thought of as opposites in the way in which forces are transferred to their intermediate nodes. Free spacers only apply tangential forces in the plane of the spacer, whereas cables only apply normal forces.

Spacer Elements

Spacers are designed for soap-film form-finding only and it will be ignored in all other analyses even though it may have been defined in the model. In a soap-film, form-finding analysis, spacer elements are used to make up a multiple node super element called spacers. A spacer is used to maintain or adjust the nodal distance along the spacer element chain as desired in the form-found structure. Spacers should lie over 2D element surface or along bar or tie elements with soap-film form-finding properties (i.e. elements with zero stiffness)

All spacers can be considered as chains of bars with initial lengths being set to about half of their actual lengths (to ensure that the Spacers remain in tension under reasonable conditions), the stiffness (AE) of spacer elements are defined in spacer properties. To reduce the influence of the spacer elements on the form-found shape of the structure, the stiffness of spacer elements should be as small as possible as long as it can maintain the required nodal spacing. Depending on the spacer type, one or two components of the spacer forces may be suppressed, so spacer elements will only control the nodal spacing and not affect the form-found shape.

As some components of spacer forces are suppressed, the nodes attached to geodesic and free spacer elements are not be in equilibrium at the end of form-finding analysis. In this respect spacers differ from other elements.

There are three types of spacer elements

- Geodesic

- Free
- Bar

The type of spacer element is defined in the spacer properties. They differ in the way that the force exerted by the spacer on the internal nodes is treated. For each internal node, a vector defines which components of the resultant spacer force on the node are taken into account. This force pulls the node along the spacer vector. The remaining components are ignored.

The spacer types also differ in the way in which the spacing rules (defined as spacer leg length type) are applied within the program.

Generally geodesic spacers should be used along 2D element surface. Free spacers should be used to control the nodal spacing along in bar or tie elements. Similar to free spacers, bar spacers can be used to control nodal spacing along bar or tie elements, the difference between free and bar elements is that no component of Bar spacer element forces will be suppressed.

Bar and free spacers simply adjust the spacing of their nodes along the coincident bar or tie elements. Geodesic spacers shift their internal nodes over the surface so as to minimize the overall length between the end nodes (a geodesic is the shortest route over a surface between two points) and also maintain the spacing of their nodes.

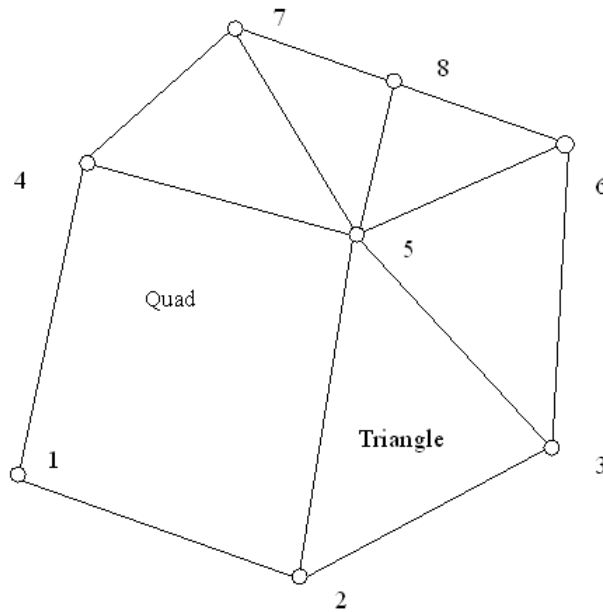
To understand the different spacer types, it is useful to consider a short length of spacer chain composed of two spacer elements jointed at the common node.

Geodesic Spacers

For geodesic spacers the component of each spacer node reaction which is normal to the surface of the adjacent soap-film triangle and quad elements is suppressed and so the spacers do not affect the form-found surface; they only control the position of nodes on the surface which is formed. The tension of the geodesic spacers ensures that the spacer nodes shift to form a geodesic upon the soap-film surface that is simultaneously formed by the triangle and quad elements.

The geodesic spacer normal is calculated as follows. The program searches for triangle and quad elements that are connected to the internal nodes of each geodesic spacer and attempts to form a rosette of elements around each node. Failure causes a program error. The nodes of the elements in the rosette are used as 'control nodes' to calculate the normal of the surface on which the spacer lies.

Initially, the normal direction is defined from the element geometry of the surrounding 2D elements. This is then adjusted during analysis based on the displacements of the surrounding nodes.



The initial normal of geodesic spacer at node 5 in the above example is equal to

$$\mathbf{n} = (\mathbf{c}_7 - \mathbf{c}_2) \times (\mathbf{c}_3 - \mathbf{c}_4) + (\mathbf{c}_8 - \mathbf{c}_3) \times (\mathbf{c}_6 - \mathbf{c}_7)$$

where the $\mathbf{c}_1 \dots \mathbf{c}_8$ are the coordinate of the nodes.

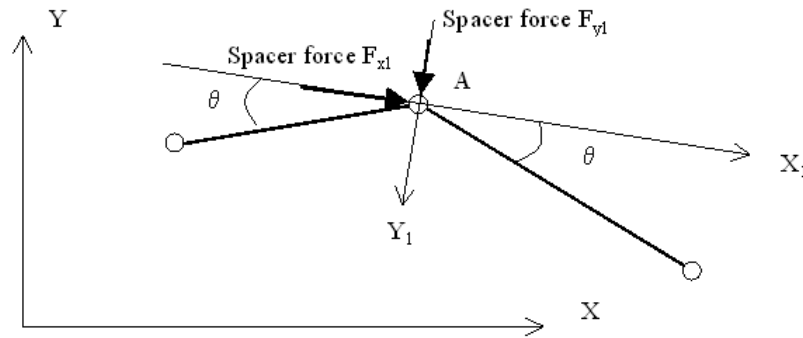
Later in the form-finding analysis, the normal will be rotated based on the displacement of the nodes. The degree of the rotation is calculated assuming all the surrounding nodes (2, 3, 4, 6, 7 and 8) are connected to the spacer node (5) by beams. The beams are pin-connected at the outer side and fixed at node 5. The EI of the beams is proportional to the sum of the angles each side of the beam. The fixed end moment for each beam at the end of node 5 is calculated. The resultant of the moment at node 5 will rotate node 5, and this rotation will be the one that rotates the spacer normal at node 5.

The normals of the end nodes of a Geodesic Spacer are found (if required) by first computing the normal as for an internal node except that

- The rosette of triangles and quads may be incomplete.
- The current normal is only rotated about an x axis lying in the plane of the normal and the end spacer leg.

The normal of the next internal node is then projected onto the plane of the rotated end normal and the end spacer leg, and reflected about the plane normal to the end spacer leg.

Normals are calculated every 10 cycles.



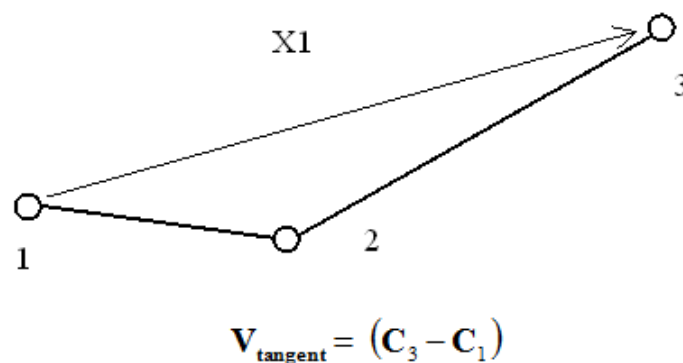
For the above portion of a geodesic spacer, assume the normal of the plane of soap film triangles and quads joined to node A is in the $z = z_1$ direction. The spacer force applied at node A is $(F_{x1}, F_{y1}, 0)$. So the spacer will move node A in the X_1 & Y_1 direction.

The geodesic spacer applies the spacer leg length rules at the start of analysis. The node positions are adjusted at the start of the analysis to meet the rules by factoring the unstressed element lengths. On convergence, the node positions may not meet the leg length rules exactly. If this proves a problem, the use of 'greasy pole' restraints or bar spacers could be considered for a final stage form-finding analysis starting from the converged form-found shape. With both options, out-of-balance forces will exist when converting from the soap film to the 'real' model, and care is needed in their interpretation.

Free Spacers

For free spacers only the component of a nodal reaction that acts along a spacer's tangent is preserved. So a free spacer only influences the spacing of nodes, along the length of a 1D form-finding element.

The initial tangent of free spacer at node 2 in the following example is equal to

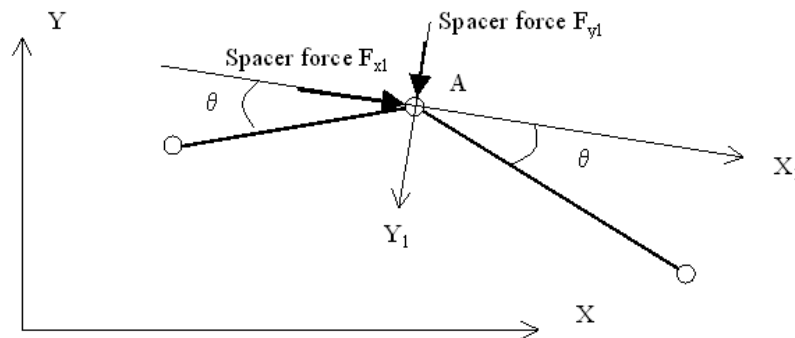


Note: $\mathbf{V}_{\text{tangent}}$ is the tangent vector at node 2 (= X_1). $\mathbf{C}_1, \mathbf{C}_3$ are coordinates of the nodes 1 and 3.

Later in the form-finding analysis, the normal will be rotated according to the displacement of the 3 nodes. The degree of the rotation is calculated assuming nodes 1 and 3 are connected to the spacer node (2) by beams. The beams are pin-connected at the nodes 1 and 3 and fixed at node 2. The EI of the two beams are equal. The fixed end moment for the two beams at the end of node 2 is calculated. The resultant of the moment at node 2 will rotate node 2, and this

rotation is the one that rotates the spacer tangent vector at node 2. The rotation will be about the normal of the plane of the adjacent spacer legs.

The tangents of the end nodes of a free spacer are found (if required) by reflecting the tangent of the next internal node about the plane normal to the end spacer leg. Tangents are calculated every 10 cycles.



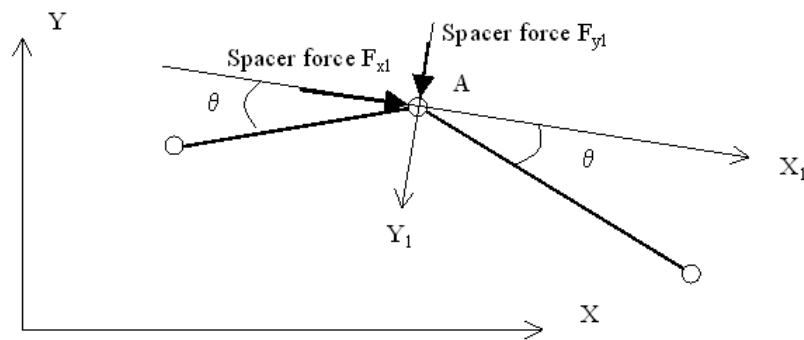
For the above portion of free spacer, assume the normal of the plane of the three spacer nodes is in the $z = z_1$ direction. The spacer force applied at node A is $(F_{x1}, 0, 0)$. So the spacer will move node A in the X_1 direction only.

The free spacer applies the spacer leg length rules at the start of analysis. The node positions are adjusted at the start of the analysis to meet the rules by factoring the unstressed element lengths. On convergence, the node positions may not meet the leg length rules exactly. If this proves to be a problem, consider replacing the free spacer with a bar spacer for a final stage form finding analysis starting from the converged form-found shape.

Bar Spacer

Bar spacer should run parallel to 1D tie elements with soap-film properties, and the two elements should have common nodes. No component of a bar spacer force is suppressed so a bar spacer influences the spacing of nodes, along the length of the soap-film tie, and the position of the soap-film tie. The soap-film tie has zero stiffness, but the bar spacer has a small stiffness. Therefore forces may vary along the bar spacer. To maintain equilibrium post form-finding, bar spacer forces should be added to the tie pre-stress in the final model as spacer elements are only considered during form-finding analysis. The effect of the bar spacer on the final form-found shape increases with the bar spacer stiffness.

The bar spacer is, in effect, a multi-noded bar super-element, or chain, with a low stiffness that complies with spacing rules for the nodes along its length (defined by the spacer leg length type)



For the above portion of Bar SPACER, assume the normal of the plane of the three spacer nodes is in the $z = z_1$ direction. The spacer force applied at node A is $(F_{x1}, F_{y1}, 0)$. So the spacer will move node A in the X_1 and Y_1 direction.

The bar spacer applies the spacer leg length rules by repeated analysis. The node positions are adjusted at the start of the analysis to meet the rules. On convergence, the node positions are adjusted along the length of the bar spacer, and the analysis repeated. This continues until the spacing rules are met.

The position of the nodes is adjusted by factoring the unstressed element lengths.

Spacer leg length type

There are three rules, or options, to control nodal spacing along a spacer, which can be selected when defining the spacer properties. These spacing rules are proportional, ratio, and projected ratio. Free and Geodesic spacers apply these rules to the initial leg length at the start of analysis but do not recheck on convergence. For spacers included in soap film structures, the ratio of the final leg length to the initial leg length will be approximately constant and the spacing rules will be met approximately. However if varying point loads or other constraints are applied along the length of the spacer the final spacing may be too approximate to be satisfactory. In this case the use of a Bar Spacer may be considered. For Bar Spacers, the spacing rules are checked on convergence, and the analysis is repeated until the leg length rules are met exactly.

The three options offered as leg length type are:

- Proportional – the final length of the spacer legs will be proportional to their original length. To achieve this, the initial length of each spacer leg is set to half the initial distance between end nodes.
- Ratio – the spacer leg length ratio will be equal to that specified, e.g. if the ratio is specified as 2, the final leg length of the 2nd element will be twice as long as the leg length of the 1st and so on. If a ratio of 1 is specified, the initial length of all the spacer legs will be made equal.
- Projected-ratio – Specify an axis set in the spacer property table. If a user defined axis set is not specified, the global axis set will be used. The spacer legs are projected onto the x-y plane of the specified axis set. The program adjusts the initial length of the spacer such

that the projected length of each spacer leg on the x-y plane follows the specified ratio rule. If a ratio of 1 is specified, the projected length of all legs will be equal. If a ratio of 2 is specified, the projected leg length of the 2nd element will be twice as long as the projected leg length of the 1st and so on.

Error Norms

When using GSA it is important that the user can be confident in the accuracy of the results. There are a number of checks that the user can carry out to check accuracy, however GSA provides some measure of the accuracy of the displacement solution. This is reported in the Analysis Details output. The definition of the error norm is different for static and modal dynamic or buckling results.

Statics

In static analysis the error norm is not calculated unless the model shows signs of being ill-conditioned. The calculation is as follows

1. Calculate the residual.
2. Solve for the displacements resulting from the residual and compare these with the actual displacements.
3. Calculate the error norm.

Thus

$$\mathbf{f}_R = \mathbf{f} - \mathbf{K}\mathbf{u}$$

$$\mathbf{u}_R = \mathbf{K}^{-1}\mathbf{f}_R$$

$$e = \|\mathbf{u}_R\|/\|\mathbf{u}\|$$

where

$$\|\mathbf{u}\| = \sqrt{\sum u_i^2}$$

Dynamics and buckling

In a dynamic analysis the error norm is always calculated as follows

$$e = \frac{\|\mathbf{K}\mathbf{u} - \lambda\mathbf{M}\mathbf{u}\|}{\|\mathbf{K}\mathbf{u}\|}$$

and in the case of buckling

$$e = \frac{\|\mathbf{K}\mathbf{u} + \lambda\mathbf{K}_g\mathbf{u}\|}{\|\mathbf{K}\mathbf{u}\|}$$

Explicit Time Integration

The explicit time integration scheme can be written as

$$a_{i,t} = \frac{f_{i,t}}{m_i}$$

$$v_{i,t+\Delta t/2} = v_{i,t-\Delta t/2} + a_{i,t}\Delta t$$

$$u_{i,t+\Delta t} = u_{i,t} + v_{i,t+\Delta t/2}\Delta t$$

The force vector at time t is the sum of all the forces acting on the nodes (degrees of freedom).

$$f_i = f_{i,app} + f_{i,int}$$

For an element the internal force vector for linear and geometrically non-linear problems is calculated from

$$\{f_{int}\} = [K]\{u_t\}$$

and if Rayleigh damping is included

$$\{f_{int}\} = [K](\{u_t\} + \beta\{v_t\}) + \alpha[M]\{v_t\}$$

2D Element Stresses and Forces

Stress in 2D Elements

Strain Definitions

The normal definitions of strain used are as follows

$$\begin{aligned} \varepsilon_{xx} &= \frac{\partial u}{\partial x} & \varepsilon_{yy} &= \frac{\partial v}{\partial y} & \varepsilon_{zz} &= \frac{\partial w}{\partial z} \\ \gamma_{xy} &= \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} & \gamma_{yz} &= \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} & \gamma_{zx} &= \frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \end{aligned}$$

An alternative definition which fits more neatly in tensor form is

$$\begin{aligned} \varepsilon_{xx} &= \frac{\partial u}{\partial x} & \varepsilon_{yy} &= \frac{\partial v}{\partial y} & \varepsilon_{zz} &= \frac{\partial w}{\partial z} \\ \varepsilon_{xy} &= \frac{1}{2} \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) & \varepsilon_{yz} &= \frac{1}{2} \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) & \varepsilon_{zx} &= \frac{1}{2} \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) \end{aligned}$$

with the strain tensor defined as

$$\varepsilon = \begin{bmatrix} \varepsilon_{xx} & \varepsilon_{xy} & \varepsilon_{xz} \\ \varepsilon_{yx} & \varepsilon_{yy} & \varepsilon_{yz} \\ \varepsilon_{zx} & \varepsilon_{zx} & \varepsilon_{zz} \end{bmatrix}$$

The calculation of principal strains $\epsilon_1, \epsilon_2, \epsilon_3$ follows from

$$E^3 - (\epsilon_{xx} + \epsilon_{yy} + \epsilon_{zz})E^2 + (\epsilon_{xx}\epsilon_{yy} + \epsilon_{yy}\epsilon_{zz} + \epsilon_{zz}\epsilon_{xx} - \epsilon_{xy}^2 - \epsilon_{yz}^2 - \epsilon_{zx}^2)E - (\epsilon_{xx}\epsilon_{yy}\epsilon_{zz} + 2\epsilon_{xy}\epsilon_{yz}\epsilon_{zx} - \epsilon_{xx}\epsilon_{yz}^2 - \epsilon_{yy}\epsilon_{zx}^2 - \epsilon_{zz}\epsilon_{xy}^2) = 0$$

The maximum shear strain is calculated from the principal strain as

$$\epsilon_{\max \text{ shear}} = \frac{1}{2}(\epsilon_1 - \epsilon_3)$$

or

$$\gamma_{\max \text{ shear}} = (\epsilon_1 - \epsilon_3)$$

In a similar way to the definitions of average and von Mises stress a volumetric and effective strain can be calculated as

$$\epsilon_{av} = \frac{1}{3}(\epsilon_{xx} + \epsilon_{yy} + \epsilon_{zz})$$

$$\epsilon_{vM} = \frac{1}{\sqrt{2}(1+\nu)} \left[(\epsilon_{xx} - \epsilon_{yy})^2 + (\epsilon_{yy} - \epsilon_{zz})^2 + (\epsilon_{zz} - \epsilon_{xx})^2 + 6(\epsilon_{xy}^2 + \epsilon_{yz}^2 + \epsilon_{zx}^2) \right]$$

Stress Definitions

Stress can be considered as a tensor quantity whose components can be represented in matrix form as

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zx} & \sigma_{zz} \end{bmatrix}$$

where each term corresponds to a force per unit area. The following notation for the stress components is common

$$\begin{aligned} \sigma_x &= \sigma_{xx} & \sigma_y &= \sigma_{yy} & \sigma_z &= \sigma_{zz} \\ \tau_{xy} &= \sigma_{xy} & \tau_{yz} &= \sigma_{yz} & \tau_{zx} &= \sigma_{zx} \end{aligned}$$

The principal stresses $\sigma_1, \sigma_2, \sigma_3$ are calculated as the roots (S) of the cubic

$$S^3 - I_1 S^2 + I_2 S - I_3 = 0$$

where the terms I_1, I_2, I_3 are stress invariants defined as

$$\begin{aligned} I_1 &= \sigma_x + \sigma_y + \sigma_z \\ I_2 &= \sigma_x \sigma_y + \sigma_y \sigma_z + \sigma_z \sigma_x - \tau_{xy}^2 - \tau_{yz}^2 - \tau_{zx}^2 \\ I_3 &= \sigma_x \sigma_y \sigma_z + 2\tau_{xy} \tau_{yz} \tau_{zx} - \sigma_x \tau_{yz}^2 - \sigma_y \tau_{zx}^2 - \sigma_z \tau_{xy}^2 \end{aligned}$$

Alternatively the principal stress equation can be written

$$\begin{aligned} S^3 - (\sigma_x + \sigma_y + \sigma_z) S^2 + \\ (\sigma_x \sigma_y + \sigma_y \sigma_z + \sigma_z \sigma_x - \tau_{xy}^2 - \tau_{yz}^2 - \tau_{zx}^2) S - \\ (\sigma_x \sigma_y \sigma_z + 2\tau_{xy} \tau_{yz} \tau_{zx} - \sigma_x \tau_{yz}^2 - \sigma_y \tau_{zx}^2 - \sigma_z \tau_{xy}^2) = 0 \end{aligned}$$

The maximum shear stress is calculated from the principal stress as

$$\tau_{\max} = \frac{1}{2}(\sigma_1 - \sigma_3)$$

Two other stress measures that are used are the average or hydrostatic stress and the von Mises stress; these are defined as

$$\sigma_{av} = \frac{1}{3}(\sigma_x + \sigma_y + \sigma_z)$$

$$\sigma_{vM} = \frac{1}{\sqrt{2}} \left[(\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2 + (\sigma_z - \sigma_x)^2 + 6(\tau_{xy}^2 + \tau_{yz}^2 + \tau_{zx}^2) \right]$$

Stress in 2D elements

The stress in 2D elements is calculated via the strains. The strains are calculated from the displacements using the strain displacement relationship **B** (see 2D elements). Using the interpolation functions these can be calculated at any point in the element. Once the strains are calculated the stress can be calculated using the material elastic matrix **C** for example for an elastic isotropic material the material matrix is

$$\mathbf{C} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & & \\ \nu & 1 & & \\ & & \frac{1-\nu}{2} & \\ & & & \frac{1-\nu}{2} \end{bmatrix}$$

Thus the strains are

$$\boldsymbol{\varepsilon} = \mathbf{B}\mathbf{u}$$

and the stresses are

$$\boldsymbol{\sigma} = \mathbf{C}\boldsymbol{\varepsilon} = \mathbf{C}\mathbf{B}\mathbf{u}$$

This can be used to evaluate the stress at any point in the element. However the stress is based on the strain which in turn is based on the displacement gradients in the element. Thus some of the strain terms in an element that has a parabolic displacement field are linear. It has been found that the best stress results are obtained by evaluating the stress at particular points (the points used for the element integration) and extrapolating the results to the nodes.

In order to have good stress results the mesh will have to be finer than the mesh required for the displacement solution and the stress results are likely to be influenced by high displacement gradients in the element.

Direct extrapolation of results

In the case of direct extrapolation a function is chosen to represent the variation of stress over the element based on the number of Gauss points. In practice this is used when there are 1, 3 or 4 Gauss points. The corresponding polynomial functions are

$$f = a$$

$$f = a + bx + cy$$

$$f = a + bx + cy + dxy$$

For 1 Gauss point the values are assumed to be constant over the whole element so the Gauss point values are simply copied to the nodes.

For 3 Gauss points the values are the Gauss points are known so the following set of equations can be set up

$$f_0 = a + bx_0 + cy_0$$

$$f_1 = a + bx_1 + cy_1$$

$$f_2 = a + bx_2 + cy_2$$

This can then be used to calculate the coefficients a, b, c

$$c = \frac{(f_0 - f_1)(x_0 - x_2) - (f_0 - f_2)(x_0 - x_1)}{(y_0 - y_1)(x_0 - x_2) - (y_0 - y_2)(x_0 - x_1)}$$

$$b = \frac{(f_0 - f_1) - c(y_0 - y_1)}{(x_0 - x_1)}$$

$$a = f_0 - bx_0 - cy_0$$

For 4 Gauss points a similar approach can be used, but in this case the locations of the Gauss points are at

$$r, s = \pm \frac{1}{\sqrt{3}}$$

so the equations can be written in the form

$$f_0 = a - bx - cy + dxy$$

$$f_1 = a + bx - cy + dxy$$

$$f_2 = a + bx + cy + dxy$$

$$f_3 = a - bx + cy - dxy$$

This can then be solved for the coefficients a, b, c, d

$$a = \frac{f_0 + f_1 + f_2 + f_3}{4}$$

$$b = \frac{-f_0 + f_1 + f_2 - f_3}{4x}$$

$$c = \frac{-f_0 - f_1 + f_2 + f_3}{4y}$$

$$d = \frac{f_0 - f_1 + f_2 - f_3}{4xy}$$

Once these are established the polynomial functions can be used to establish the values at any position on the element.

Least squares extrapolation of results

In this case a function chosen to fit through the points would imply a higher order polynomial than the one used to interpolate the geometry, so a least squares approach is used to find the

polynomial to map the stresses from the Gauss points to the nodes. The interpolation functions used for 6 node and 8 node elements are respectively

$$f = a_0 + a_1x + a_2y + a_3xy + a_4x^2 + a_5y^2$$

$$f = a_0 + a_1x + a_2y + a_3xy + a_4x^2 + a_5y^2 + a_6x^2y + a_7xy^2$$

The square of the error for any point is then

$$e^2 = (p(x, y) - f)^2$$

This is summed over all the Gauss points and then the derivatives with respect to the coefficients are set to zero (selecting the coefficients that minimise the error). This leads to the matrix equation for 8 node elements

$$\begin{bmatrix} n & \sum x & \sum y & \sum xy & \sum x^2 & \sum y^2 & \sum x^2y & \sum xy^2 \\ & \sum x^2 & \sum xy & \sum x^2y & \sum x^3 & \sum xy^2 & \sum x^3y & \sum x^2y^2 \\ & & \sum y^2 & \sum xy^2 & \sum x^2y & \sum y^3 & \sum x^2y^2 & \sum xy^3 \\ & & & \sum x^2y^2 & \sum x^3y & \sum xy^3 & \sum x^3y^2 & \sum x^2y^3 \\ & & & & \sum x^4 & \sum x^2y^2 & \sum x^4y & \sum x^3y^2 \\ & & & & & \sum y^4 & \sum x^2y^3 & \sum xy^4 \\ & & & & & & \sum x^4y^2 & \sum x^3y^3 \\ & & & & & & & \sum x^2y^4 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \end{Bmatrix} = \begin{Bmatrix} \sum f \\ \sum fx \\ \sum fy \\ \sum fxy \\ \sum fx^2 \\ \sum fy^2 \\ \sum fx^2y \\ \sum fxy^2 \end{Bmatrix}$$

The 6 node version is the same except that the a_6 and a_7 terms are ignored. This can then be solved for the coefficients.

Stress-strain Relationships

The relationship between stress and strain depends on the type of problem

Problem	Displacements	Strain	Stress
Plane stress	u, v	$\epsilon_{xx}, \epsilon_{yy}, \epsilon_{xy}$	$\sigma_{xx}, \sigma_{yy}, \sigma_{xy}$
Plane strain	u, v	$\epsilon_{xx}, \epsilon_{yy}, \epsilon_{xy}$	$\sigma_{xx}, \sigma_{yy}, \sigma_{xy}$
Axisymmetric	u, v	$\epsilon_{xx}, \epsilon_{yy}, \epsilon_{xy}$	$\sigma_{xx}, \sigma_{yy}, \sigma_{xy}$
Plate Bending	w	$\kappa_{xx}, \kappa_{yy}, \kappa_{xy}$	M_{xx}, M_{yy}, M_{xy}
General	u, v, w	$\epsilon_{xx}, \epsilon_{yy}, \epsilon_{zz}, \epsilon_{xy}, \epsilon_{yz}, \epsilon_{zx}$	$\sigma_{xx}, \sigma_{yy}, \sigma_{zz}, \sigma_{xy}, \sigma_{yz}, \sigma_{zx}$

Where for beam and plate bending problems the relationship is between moment and curvature

$$\kappa_{xx} = -\frac{\partial^2 w}{\partial x^2} \qquad \kappa_{yy} = -\frac{\partial^2 w}{\partial y^2} \qquad \kappa_{xy} = -\frac{\partial^2 w}{\partial x \partial y}$$

The stress-strain matrices for isotropic materials are

Problem	Stress-strain matrix
Plane stress	$\frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}$
Plane strain	$\frac{E(1-\nu)}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1 & \frac{\nu}{1-\nu} & 0 \\ \frac{\nu}{1-\nu} & 1 & 0 \\ 0 & 0 & \frac{1-2\nu}{2(1-\nu)} \end{bmatrix}$
Axisymmetric	$\frac{E(1-\nu)}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1 & \frac{\nu}{1-\nu} & \frac{\nu}{1-\nu} & 0 \\ \frac{\nu}{1-\nu} & 1 & \frac{\nu}{1-\nu} & 0 \\ \frac{\nu}{1-\nu} & \frac{\nu}{1-\nu} & 1 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2(1-\nu)} \end{bmatrix}$
Plate bending	$\frac{Et^3}{12(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}$
General	$\frac{E(1-\nu)}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1 & \frac{\nu}{1-\nu} & \frac{\nu}{1-\nu} & 0 & 0 & 0 \\ \frac{\nu}{1-\nu} & 1 & \frac{\nu}{1-\nu} & 0 & 0 & 0 \\ \frac{\nu}{1-\nu} & \frac{\nu}{1-\nu} & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2(1-\nu)} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2(1-\nu)} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2(1-\nu)} \end{bmatrix}$

The stress-strain matrices for orthotropic materials:

Problem	Stress-strain matrix
Plane stress	$\frac{1}{1-\nu_{xy}\nu_{yx}} \begin{bmatrix} E_x & \nu_{xy}E_y & 0 \\ & E_y & 0 \\ \text{symmetric} & & G_{xy} \end{bmatrix}$

$$\text{Plane strain} \quad \begin{bmatrix} \frac{E_x(1-\nu_{yz}\nu_{zy})}{D} & \frac{E_y(\nu_{xy}-\nu_{zy}\nu_{xz})}{D} & 0 \\ \text{symmetric} & \frac{E_x(1-\nu_{xz}\nu_{zx})}{D} & 0 \\ & & G_{xy} \end{bmatrix}$$

Where

$$D = 1 - \nu_{xy}\nu_{yx} - \nu_{zx}(\nu_{xy}\nu_{yz} + \nu_{xz}) - \nu_{zy}(\nu_{xz}\nu_{yx} + \nu_{yz})$$

$$\text{Axisymmetric} \quad \begin{bmatrix} \frac{1}{E_x} & \frac{-\nu_{yx}}{E_y} & \frac{-\nu_{zx}}{E_z} & 0 \\ & \frac{1}{E_y} & \frac{-\nu_{zy}}{E_z} & 0 \\ & & \frac{1}{E_z} & 0 \\ \text{symmetric} & & & \frac{1}{G_{xy}} \end{bmatrix}^{-1}$$

$$\text{Plate bending} \quad \frac{t^3}{12} \begin{bmatrix} \frac{E_x}{1-\nu_{xy}\nu_{yx}} & \frac{\nu_{yx}E_y}{1-\nu_{xy}\nu_{yx}} & 0 \\ \text{symmetric} & \frac{E_y}{1-\nu_{xy}\nu_{yx}} & 0 \\ & & G_{xy} \end{bmatrix}$$

$$\text{General} \quad \begin{bmatrix} \frac{1}{E_x} & \frac{-\nu_{yx}}{E_y} & \frac{-\nu_{zx}}{E_z} & 0 & 0 & 0 \\ & \frac{1}{E_y} & \frac{-\nu_{zy}}{E_z} & 0 & 0 & 0 \\ & & \frac{1}{E_z} & 0 & 0 & 0 \\ & & & \frac{1}{G_{xy}} & 0 & 0 \\ & & & & \frac{1}{G_{yz}} & 0 \\ \text{symmetric} & & & & & \frac{1}{G_{xz}} \end{bmatrix}^{-1}$$

Force in 2D Elements

The Timoshenko convention is used for forces in 2D elements. This means that a moment M_x is based on the stress in the x direction. With the Timoshenko convention if a slab is in compression on the top face in both the x and y directions the moments are both negative.

Consequently starting from the assumption that tensile stress is positive, we have the following relationships for the forces

$$N_x = \int_{-t/2}^{t/2} \sigma_{xx} dz \quad N_y = \int_{-t/2}^{t/2} \sigma_{yy} dz \quad N_{xy} = \int_{-t/2}^{t/2} \sigma_{xy} dz$$

$$Q_x = \int_{-t/2}^{t/2} \sigma_{xz} dz \quad Q_y = \int_{-t/2}^{t/2} \sigma_{yz} dz$$

and moments

$$M_x = \int_{-t/2}^{t/2} \sigma_{xx} z dz \quad M_y = \int_{-t/2}^{t/2} \sigma_{yy} z dz \quad M_{xy} = \int_{-t/2}^{t/2} \sigma_{xy} z dz$$

Following from this a plate that has a positive in-plane stress in x/y will have a positive force resultant and a positive bending stress in x/y (i.e. positive stress at the top surface relative to the bottom surface) will have a positive moment.

When the structure is linear these simplify to:

$$N_x = \sigma_{p,xx} t \quad N_y = \sigma_{p,yy} t \quad N_{xy} = \sigma_{p,xy} t$$

$$Q_x = \sigma_{xz} t \quad Q_y = \sigma_{yz} t$$

and

$$M_x = \sigma_{b,xx} \frac{t^2}{6} \quad M_y = \sigma_{b,yy} \frac{t^2}{6} \quad M_{xy} = \sigma_{b,xy} \frac{t^2}{6}$$

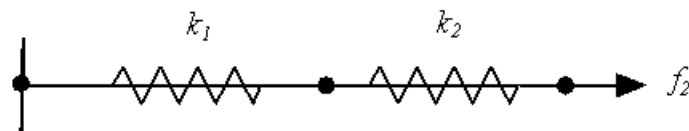
where the superscripts p and b refer to in-plane and bending stress terms.

When in-plane and bending thickness modifiers are used the in-plane forces are based on the in-plane thickness and the moments and shear forces are based on the bending thickness. Stresses are always based on the actual thickness of the element.

III Conditioning

In the vast majority of cases the solver will give a correct solution to the problem. However, some problems are by nature ill-conditioned in which case small changes in the input data can lead to more significant changes in the results.

Taking a simple example to look at ill-conditioning; consider a simple two spring system, where the springs are connected in series. The stiffness of the first spring is k_1 and that of the second is k_2 and we assume that k_2 is much greater than k_1 ($k_2 \gg k_1$).



In this case the equations describing the system is

$$\begin{Bmatrix} 0 \\ f_2 \end{Bmatrix} = \begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix}$$

As in a solver based on a Gaussian elimination technique, we use these equations to arrive at a relationship between u_2 and u_1

$$u_2 = \frac{f_2 + k_2 u_1}{k_2}$$

which when substituted in the other equation gives:

$$(k_1 + k_2)u_1 - k_2 \left(\frac{f_2 + k_2 u_1}{k_2} \right) = 0$$

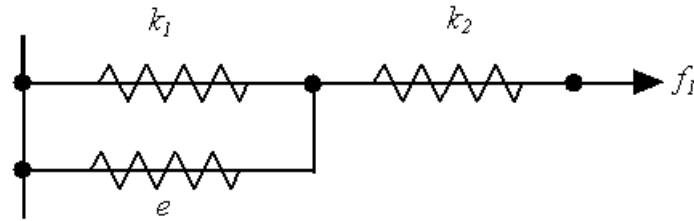
or

$$[k_1 + (k_2 - k_2)]u_1 = f_2$$

With exact arithmetic the term $(k_2 - k_2)$ would be zero, however, if k_2 is large compared with k_1 and due to limited precision, some error will be introduced in the calculation. If this error is denoted by e , then the equation we have is

$$[k_1 + e]u_1 = f_2$$

We have then a system as shown below where the error is like adding a third spring, which acts in parallel with k_1 .



The expected reaction is f_2 , but the reaction that is calculated is

$$r = k_1 u_1 = \frac{k_1 f_2}{k_1 + e}$$

Thus the reaction is in error by a factor

$$\frac{k_1}{k_1 + e}$$

Interpolation on a Triangular/Quad Facet

The r, s coordinates of a triangular facet can be determined from the use of interpolation (shape) functions. Let

$$\begin{aligned} u_i &= (1 - r - s)u_1 + ru_2 + su_3 \\ v_i &= (1 - r - s)v_1 + rv_2 + sv_3 \end{aligned}$$

These can be rewritten as

$$\begin{aligned} (u_i - u_1) &= (u_2 - u_1)r + (u_3 - u_1)s \\ (v_i - v_1) &= (v_2 - v_1)r + (v_3 - v_1)s \end{aligned}$$

and then

$$\begin{aligned} r &= \frac{(v_3 - v_1)(u_i - u_1) - (u_3 - u_1)(v_i - v_1)}{(v_3 - v_1)(u_2 - u_1) - (u_3 - u_1)(v_2 - v_1)} \\ s &= \frac{(u_i - u_1) - (u_2 - u_1)r}{(u_3 - u_1)} \end{aligned}$$

Then

$$f_i = (1 - r - s)f_1 + rf_2 + sf_3$$

On a quadrilateral facet using the interpolation functions gives

$$\begin{aligned} u_i &= \frac{1}{4}(1 - r)(1 - s)u_1 + \frac{1}{4}(1 + r)(1 - s)u_2 + \frac{1}{4}(1 + r)(1 + s)u_3 + \frac{1}{4}(1 - r)(1 + s)u_4 \\ v_i &= \frac{1}{4}(1 - r)(1 - s)v_1 + \frac{1}{4}(1 + r)(1 - s)v_2 + \frac{1}{4}(1 + r)(1 + s)v_3 + \frac{1}{4}(1 - r)(1 + s)v_4 \end{aligned}$$

these can be rewritten

$$u_i = u_a + u_b r + u_c s + u_d r s$$

$$v_i = v_a + v_b r + v_c s + v_d r s$$

Using the first of these gives

$$s = \frac{((u_i - u_a) - u_b r)}{(u_c + u_d r)}$$

which can be substituted into the second to give

$$v_i = v_a + v_b r + (v_c + v_d r) \frac{((u_i - u_a) - u_b r)}{(u_c + u_d r)}$$

This can then be solved for r & s and then the interpolation function used as for the triangular facet.

Loading

Beam Loads

The reference mechanical load is the point load; all the other mechanical load types can be established by integrating the results for a point load over the loaded part of the beam.

The basic approach to calculating the load on the beam for a force at position a is to consider the beam split at a into two separate beams. Flexibility matrices can be established for axial, torsional and flexural loading

$$f_{ax} = \frac{l}{EA} \quad f_{tor} = \frac{l}{GJ}, \quad f_{flex} = \frac{1}{EI} \begin{bmatrix} \frac{l}{4} & \frac{l^2}{4} \\ \frac{l^2}{6} & \frac{l^3}{12} + \frac{EI}{GA_s} \end{bmatrix}$$

There must be continuity of displacement and rotation between the two beams and the forces and moment must balance the applied load. This allows a set of equations to be set up for the sub-beams a and b which can be solved for the shear force and bending moment at the loaded point.

$$f_a w_a = f_b w_b$$

Where the vector W is respectively for unit force and unit moment

$$w_f = \begin{Bmatrix} 1 \\ 0 \end{Bmatrix} \quad w_m = \begin{Bmatrix} 0 \\ 1 \end{Bmatrix}$$

Once the force and moment at the loaded point have been established the end forces and moments (and hence the equivalent nodal forces) result from equilibrium of the two sub-beams.

The general distributed loading in the patch load, varying in linearly in intensity from position a to position b . The nodal forces and moments are then given by integrating the results for a point load

$$f = \int_a^b (w_1 + w_2 x) f_p(x) dx$$

where $f_p(x)$ is the force due to a point load at x and

$$w_1 = \frac{w_a b - w_b a}{b - a} \quad w_2 = \frac{w_b - w_a}{b - a}$$

The tri-linear load option is simply a repeated set of patch loads.

Beam Thermal Loads

The thermal loads allow for the introduction of load due to temperature variations in the beam elements. For an axial thermal loading the expansion of the element leads to axial forces, for thermal gradients through the thickness of the element both axial forces and end moments are induced.

$$\begin{aligned} f_x &= EA \alpha T_{NA} \\ m_{yy} &= EI \alpha \frac{dT}{dz} \\ m_{zz} &= EI \alpha \frac{dT}{dy} \end{aligned}$$

where α is the temperature coefficient of expansion, T is temperature and the subscript NA refers to the neutral axis.

Beam Pre-stress Loads and Lack-of-fit

Pre-stress and lack of fit loads are similar, but the definition differs. A pre-stress is considered as a force applied to the element at some position in the section relative to the neutral axis. A lack of fit is assumed to affect only the axial terms relating to the element. For a pre-stress of force and offsets of y and z the resulting nodal forces and moments are

$$\begin{aligned} f_x &= f_p \\ m_{yy} &= f_p z \\ m_{zz} &= -f_p y \end{aligned}$$

For the lack of fit and the initial strain the nodal forces are respectively

$$\begin{aligned} f_x &= \frac{EA}{l} \Delta l \\ f_x &= EA \varepsilon \end{aligned}$$

Loads on pin-ended beams

When the ends of beam are not fully fixed, some adjustment has to be made to the forces and moments that are applied at the nodes and the pinned end cannot sustain any moment. The simplest case is when both ends are pinned so

$$m_1 = m_2 = 0$$

The forces and moments are then modified to maintain equilibrium as follows

$$\begin{aligned} f_1 &\rightarrow f_1 \pm \frac{(m_1 + m_2)}{l}, & m_1 &\rightarrow 0 \\ f_2 &\rightarrow f_2 \mp \frac{(m_1 + m_2)}{l}, & m_2 &\rightarrow 0 \end{aligned}$$

for y/z.

When the element is pinned at one end only the corrections depend on the material properties in the general case

pin at end 1

$$f_1 \rightarrow f_1 \pm \left(\frac{6}{4 + \alpha} \right) \frac{m_1}{l}$$

$$f_2 \rightarrow f_2 \mp \left(\frac{6}{4 + \alpha} \right) \frac{m_1}{l}$$

$$m_1 \rightarrow 0$$

$$m_2 \rightarrow m_2 - \left(\frac{2 - \alpha}{4 + \alpha} \right) m_1$$

pin at end 2

$$f_1 \rightarrow f_1 \mp \left(\frac{6}{4 + \alpha} \right) \frac{m_2}{l}$$

$$f_2 \rightarrow f_2 \pm \left(\frac{6}{4 + \alpha} \right) \frac{m_2}{l}$$

$$m_1 \rightarrow m_1 - \left(\frac{2 - \alpha}{4 + \alpha} \right) m_2$$

$$m_2 \rightarrow 0$$

where

$$\alpha = 12 \frac{EI}{GA_s l} \quad A_s = k A$$

For a simple beam this reduces to

pin at end 1

$$f_1 \rightarrow f_1 \pm \frac{3m_1}{2l}$$

$$f_2 \rightarrow f_2 \mp \frac{3m_1}{2l}$$

$$m_1 \rightarrow 0$$

$$m_2 \rightarrow m_2 - \frac{m_1}{2}$$

pin at end 2

$$f_1 \rightarrow f_1 \mp \frac{3m_2}{2l}$$

$$f_2 \rightarrow f_2 \pm \frac{3m_2}{2l}$$

$$m_1 \rightarrow m_1 - \frac{m_2}{2}$$

$$m_2 \rightarrow 0$$

Projected Loads

When the load on a beam is distributed the total load can be based on the actual length of the element or on the projected length of the element. Thus a load normal to the x axis of the element would have a projected length equal to the beam length while a load parallel to the x axis of the beam would have a projected length of zero. In the general case of a load in the a direction defined by a unit vector \mathbf{W} and where the beam load x axis is denoted \mathbf{x} the factor that has to be applied to the load intensity, defined in terms of the angle between the two vectors, is

$$f \rightarrow f \sin \theta = f \sqrt{1 - (\mathbf{x} \cdot \mathbf{w})^2}$$

2D Body Loads

The most common form of body loads are gravity loads. The general expression for body loads is given by

$$\mathbf{r}_b = \int_b \mathbf{h}^T \mathbf{f}_b dV$$

where integration is over the volume of the element. Using Gaussian integration this can be expressed as

$$\mathbf{r}_b = \sum_i \alpha_i \mathbf{h}_i^T \mathbf{f}_{b,i} \det J_i$$

Where α are the Gaussian weights and $\det J$ is the determinant of the Jacobian.

2D Face Loads

Face loads can be constant or varying across the face of an element. For a point load on the face the load vector is

$$\mathbf{r}_s = \mathbf{h}^T f_s$$

This can be generalised for the distributed case giving the load vector as

$$\mathbf{r}_s = \int_s \mathbf{h}^T \mathbf{f}_s dS$$

where integration is over the surface of the element. Using Gaussian integration this can be expressed as

$$\mathbf{r}_s = \sum_i \alpha_i \mathbf{h}_i^T \mathbf{f}_{s,i} \det J_{s,i}$$

Where α are the Gaussian weights and $\det J$ is the determinant of the Jacobian.

2D Element Thermal Loads

Thermal loads can be either a constant temperature rise in the whole element or a temperature gradient, varying over the surface element. To evaluate the equivalent nodal forces the temperatures have to be converted to strains, using the temperature coefficient of expansion. The strains are then related to the stress through the material matrix and then the internal stresses are integrated over the element. The thermal effects are the same in all directions so there are no shear strains introduced.

For in-plane effects (constant temperature) the strain is

$$\boldsymbol{\varepsilon}_p = \begin{bmatrix} \varepsilon_{xx} & \varepsilon_{yy} & \gamma_{xy} \end{bmatrix} = \begin{bmatrix} \alpha T & \alpha T & 0 \end{bmatrix}$$

For bending effects (temperature gradients) the strain and stress are

$$\boldsymbol{\kappa}_b = \begin{bmatrix} \kappa_{xx} & \kappa_{yy} & \kappa_{xy} \end{bmatrix} = \frac{1}{t} \begin{bmatrix} 4\alpha T & 4\alpha T & 0 \end{bmatrix}$$

These are converted to stresses through the material matrix

$$\sigma_p = C_p \varepsilon_p \quad \sigma_b = C_b \kappa_b$$

2D In-plane Loads

In-plane loads break down into two categories, pre-stress loads and initial strains. Pre-stress loads are defined as a force per unit width in either x, y or both x and y directions, along with an offset from the neutral axis. Initial strains are defined in either x, y or both x and y directions, but no offsets are permitted. The pre-stress loads can be converted directly to a set of element stresses, which are integrated over the element to get the nodal forces. The offset force gives rise to a moment on the element so

$$\sigma_p = \frac{F}{t} \quad \sigma_b = \frac{6M}{t^2} = \frac{6Fz}{t^2}$$

The initial strains can be converted to stresses in a manner analogous to that used for thermal loads.

Grid Loads

Grid loading is loading applied in space by means of a grid plane. There are three basic types of grid loading: point loads, line loads and area loads. The area loads can be subdivided into loads on the whole grid plane and those on a defined area bounded by a polyline defining a closed polygon. The panels to which the loads are applied can be one-way spanning or two-way spanning or multi-way spanning.

Grid Cells

A loading grid is used to integrate area loads and for distribution of grid point loads on to beam elements around the panel. The grid used needs to be fine enough to give an adequate representation of the load, so it needs to be based on the size of the panels that are loaded. The size and shape of the panels can vary significantly, so a robust way of determining the grid size is required.

For a square panel the load can probably be represented adequately by a 4 × 4 grid, but for a long thin panel the same grid would be unsuitable. The grid size is established as follows:

Calculate the area of the panel and set a representative panel dimension to be the square root of this. Then the grid size is this value divided by the grid refinement factor. This defaults to give typically 4 cells along the edge of a square panel. The user can adjust the grid refinement factor to a lower or higher value if required.

For a series of aspect ratios with a refinement factor of 4 the mesh densities are as follows.

Aspect ratio	Cell density
1 : 1	4 × 4
1 : 2	2.83 × 5.65
1 : 4	2 × 8

1 : 10	1.265 × 12.65
1 : 16	1 × 16

The calculation of the loading grid size can then be calculated on a panel by panel basis and the final size selected to give adequate representation on the smaller panels, with not being skewed unduly by a few very small panels. To ensure that this is the case the average, μ , and standard deviation, σ , of the individual panel loading grid sizes can be calculated and the loading grid size set to $\mu - \sigma$.

Grid Point Loads

The way in which the loads are applied depends on the type of structure as represented by the span type.

One way spanning loads are calculated by assuming the load applied to a 'plank' spanning from one side of the panel to the other in the span direction. In the case of loading over the whole panel this means that the load per 'plank' is the product of the load intensity and the plank length, split evenly between each end. The algorithm replaces the plane by a line with a load intensity applied to each end of the line.

The starting point for the two way distribution of load is to consider a circle of unit radius centred on the load point. The actual intensity at the edges is calculated by extrapolating from this point using a $1/r$ function where r is the distance from the load point to the edge of the panel.

We can satisfy these requirements with a distribution of the form

$$f = 1 + A \cos(2\theta + \phi)$$

We require that there is force and moment equilibrium. The form of these functions satisfies the moment equilibrium requirement and we can look for a solution for an arbitrary load. The term

$$\cos(2\theta + \phi)$$

takes account of the aspect ratio of the panel in determining the split between the long and short directions. If the panel is square we expect the A coefficient to be 0 and 1 for an infinitely long panel. To define the coefficient and phase angle we define a length direction for the panel and a width direction. The length is chosen to be in the direction of the sum or difference of the longest diagonals, whichever has the greater magnitude. The width direction is normal to the length direction. For a rectangular panel this accords with the normal definition of length and width.

We then define values as follows:

l_{\max}, l_{\min} – maximum and minimum dimension from point to panel edge in length direction

w_{\max}, w_{\min} – maximum and minimum dimension from point to panel edge in width direction

α – the angle of the width direction from the grid x axis

then we can calculate the coefficients from

$$A = \begin{cases} 1 - \frac{|w_{\min}|}{l_{\min}} & |w_{\min}| \leq |l_{\min}| \\ 1 - \frac{l_{\min}}{|w_{\min}|} & |w_{\min}| > |l_{\min}| \end{cases}$$

$$\phi = \begin{cases} 2\alpha & |w_{\min}| \leq |l_{\min}| \\ 2\alpha + \pi & |w_{\min}| > |l_{\min}| \end{cases}$$

The load intensity at the edge of the panel is then calculated from the distance of the point on the edge from the load point.

$$p = \frac{f}{r}$$

If we consider a circle at unit radius from the grid point load we have a load intensity function of the form

$$f = 1 + A \cos(2\theta + \phi)$$

This must be mapped on to the surrounding elements. We can use the grid cell size, c , when establishing the size of the loaded patches on elements around the panel boundary.

The distribution of point loads can then be determined at a series of point that are a distance c apart along the elements on the boundary with a minimum of a start and end point on each boundary element. The number of segments for the load distribution can then be determined from the grid cell size and element length, l

$$n = \text{ceil}\left(\frac{l}{c}\right)$$

For the default value of grid load refinement and uniform sized square panels this will give four load patches along each side of the panel. The length of the patches is then

$$\Delta l = \frac{l}{n}$$

Consider two lines from the load point to the start and finish of the element segment. These will be at angles θ_0 and θ_1 . The load carried by this segment must then be

$$F_i = W \int_{\theta_0}^{\theta_1} f d\theta$$

where W is the load intensity. The angle at which this applies is determined from

$$M_i = W \int_{\theta_0}^{\theta_1} f\theta d\theta$$

$$\theta = \frac{M_i}{F_i}$$

If the vectors V_0 and v_1 are vectors from the load point to the ends of the beam segment then the grid load F_i can be thought of as a point load along the vector v_F , at an angle θ_F from V_0 .

The loading function contains terms of the form

$$\int \theta \cos(n\theta + \phi) d\theta$$

which integrates to give

$$\frac{\theta}{2} \sin(n\theta + \phi) + \frac{1}{n^2} \cos(n\theta + \phi)$$

The load F_i

$$\begin{aligned} \frac{F_i}{W} &= \int_{\theta_0}^{\theta_1} f d\theta \\ &= \int_{\theta_0}^{\theta_1} (1 + A \cos(2\theta + \phi)) d\theta \\ &= (\theta_1 - \theta_0) + \frac{A}{2} [\sin(2\theta_1 + \phi) - \sin(2\theta_0 + \phi)] \end{aligned}$$

and load moment M_i

$$\begin{aligned} \frac{M_i}{W} &= \int_{\theta_0}^{\theta_1} f\theta d\theta \\ &= \int_{\theta_0}^{\theta_1} (1 + A \cos(2\theta + \phi))\theta d\theta \\ &= \frac{(\theta_1^2 - \theta_0^2)}{2} + A \left[\left(\frac{\theta_1}{2} \sin(2\theta_1 + \phi) + \frac{1}{4} \cos(2\theta_1 + \phi) \right) - \left(\frac{\theta_0}{2} \sin(2\theta_0 + \phi) + \frac{1}{4} \cos(2\theta_0 + \phi) \right) \right] \end{aligned}$$

This point load must then be adjusted to allow for the distance of the beam from the load point. This can use a $1/r$ factor to preserve moment equilibrium. This 'point' load can then be represented as a linearly varying patch load along the segment length. If the projection of the Load point on to the element is at position l along the segment and the ends are i and j then the equilibrium conditions require that

$$F_i = \frac{(f_i + f_j)\Delta l}{2}$$

$$F_i a l = \frac{f_i \Delta l^2}{2} + \frac{(f_i - f_j)\Delta l^2}{3}$$

Knowing a these equations can be solved for f_i and f_j .

$$f_i = \frac{(6a - 2)F}{\Delta l}$$

$$f_j = \frac{2F}{\Delta l} - f_i$$

Grid Area Loads

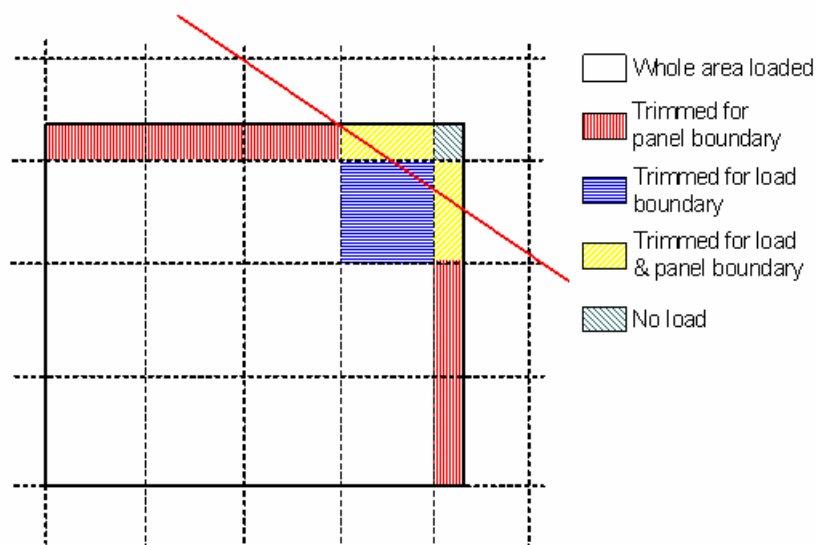
When a grid area load is applied in a multi-spanning panel the approach is to consider the loading to be represented as a grid of "point loads" distributed over the loaded area. The distributed load is then accounted for by summing all the "point loads".

Having established the loading grid size then we can then work through the grid and determine if a cell is loaded or not.

Where a cell is bisected by the load boundary then the load intensity is reduced in proportion to the loaded area and adjustment is made for the position of the load. Where a cell is bisected by the structure boundary the point of application of the load must be moved so that the load is applied to the "structure" and not in "space". In these case the centroid of the trimmed loading grid cell is calculated and the (reduced) load on the whole cell applied at the recalculated centroid.

Where a cell is bisected by a panel edge the load is applied to the panel at the centroid of the grid cell.

For the cases where this is too coarse the grid refinement factor can be increased.



Grid area loads can be projected. When this is the case the load intensity is reduced depending on the projected area of the panel to the loading axis. If the normal direction of the load is \mathbf{n}_l and the normal direction to the panel is \mathbf{n}_p . Then the load intensity is adjusted by

$$\mathbf{n}_l \cdot \mathbf{n}_p$$

When the load is not projected then the intensity depend on the panel area relative to the panel area projected on to the grid plane (normal \mathbf{n}_g) so the intensity has to be modified by

$$\frac{1}{\mathbf{n}_g \cdot \mathbf{n}_p}$$

When the panels lie in the grid plane this factor is unity.

Note: If a polygon which results in a loaded cell being split into two regions, the whole cell is assumed to be loaded. This results in a (slight) overall increase in the applied load. This restriction is in order to keep the load distribution as fast as possible.

Grid Line Loads

Grid line loads are treated in a similar manner to grid area loads in that they are broken down into a series of grid point loads along the length of the line. The same grid cell size is used to determine the number of segments along the line and thereafter the procedure is the same as for area loads.

Lagrange Interpolation

Lagrange interpolation² gives a way of fitting a polynomial through a set of points. The basic polynomial is

$$P(x) = \sum_{j=1}^n P_j(x)$$

where

$$P_j(x) = y_j \prod_{k=1, k \neq j}^n \frac{x - x_k}{x_j - x_k}$$

In order to generate a curve in space it is convenient to consider this as a four dimensional problem with parameter t as the independent variable. This ensures that t is monotonic and avoids singularities. The resulting modified equations are

² Archer, Branden and Weisstein, Eric W. "Lagrange Interpolating Polynomial." From MathWorld-- A Wolfram Web Resource. <http://mathworld.wolfram.com/LagrangeInterpolatingPolynomial.html>

$$\mathbf{P}(t) = \sum_{j=1}^n \mathbf{P}_j(t)$$

where

$$\mathbf{P}_j(t) = \mathbf{x}_j \prod_{k=1, k \neq j}^n \frac{t - t_k}{t_j - t_k}$$

For convenience the t values are assumed to be in the range [0:1]

Mass Distribution

The calculation of the mass and inertia of the structure are as follows

$$m = \sum m_i$$

$$x_c = \frac{\sum (m_i x_i)}{m}$$

$$\mathbf{I} = \begin{bmatrix} \sum I_{xx} + \sum m_i (y_i^2 + z_i^2) & \sum I_{xy} + \sum m_i (x_i y_i) & \sum I_{xz} + \sum m_i (x_i z_i) \\ \sum I_{xy} + \sum m_i (x_i y_i) & \sum I_{yy} + \sum m_i (z_i^2 + x_i^2) & \sum I_{yz} + \sum m_i (y_i z_i) \\ \sum I_{xz} + \sum m_i (x_i z_i) & \sum I_{yz} + \sum m_i (y_i z_i) & \sum I_{zz} + \sum m_i (x_i^2 + y_i^2) \end{bmatrix}$$

where the summations are over all the nodes and x, y, z are the coordinates of the node relative to the centre of mass.

If the mass option is set to ignore the element mass, this calculation is only carried out over the mass elements. If an additional mass due to load is set the load vector resulting from the load description is calculated and the required component is extracted scaled and converted to mass

$$m_i = s \frac{f_{i,j}}{g}$$

where s is the scale factor, j is the specified component and g is the gravity value.

Material Models

Isotropic material

The material properties are

E - Young's modulus

G - shear modulus

ν - Poisson's ratio

The general elasticity matrix is

$$C^{-1} = \begin{bmatrix} \frac{1}{E} & \frac{-\nu}{E} & \frac{-\nu}{E} & 0 & 0 & 0 \\ \frac{-\nu}{E} & \frac{1}{E} & \frac{-\nu}{E} & 0 & 0 & 0 \\ \frac{-\nu}{E} & \frac{-\nu}{E} & \frac{1}{E} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G} \end{bmatrix}$$

The shear modulus is related to the Young's modulus and Poisson's ratio through

$$G = \frac{E}{2(1+\nu)}$$

Orthotropic material

The material properties are

E_x - Young's modulus in the x direction

E_y - Young's modulus in the y direction

E_z - Young's modulus in the z direction

G_{xy} - shear modulus in the xy plane

G_{yz} - shear modulus in the yz plane

G_{zx} – shear modulus in the zx plane

ν_{xy} – Poisson’s ratio, y direction strain generated by unit strain in the x direction

ν_{yz} – Poisson’s ratio, z direction strain generated by unit strain in the y direction

ν_{zx} – Poisson’s ratio, x direction strain generated by unit strain in the z direction

The other three Poisson's ratios ($\nu_{yx}, \nu_{zy}, \nu_{xz}$) can be obtained from the following relationships.

$$\frac{E_x}{E_y} = \frac{\nu_{xy}}{\nu_{yx}} \quad \frac{E_y}{E_z} = \frac{\nu_{yz}}{\nu_{zy}} \quad \frac{E_z}{E_x} = \frac{\nu_{zx}}{\nu_{xz}}$$

The general elasticity matrix is

$$C^{-1} = \begin{bmatrix} \frac{1}{E_x} & \frac{-\nu_{yx}}{E_y} & \frac{-\nu_{zx}}{E_z} & 0 & 0 & 0 \\ \frac{-\nu_{xy}}{E_x} & \frac{1}{E_y} & \frac{-\nu_{zy}}{E_z} & 0 & 0 & 0 \\ \frac{-\nu_{xz}}{E_x} & \frac{-\nu_{yz}}{E_y} & \frac{1}{E_z} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{G_{yz}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{G_{zx}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{G_{xy}} \end{bmatrix}$$

The von Mises stress is

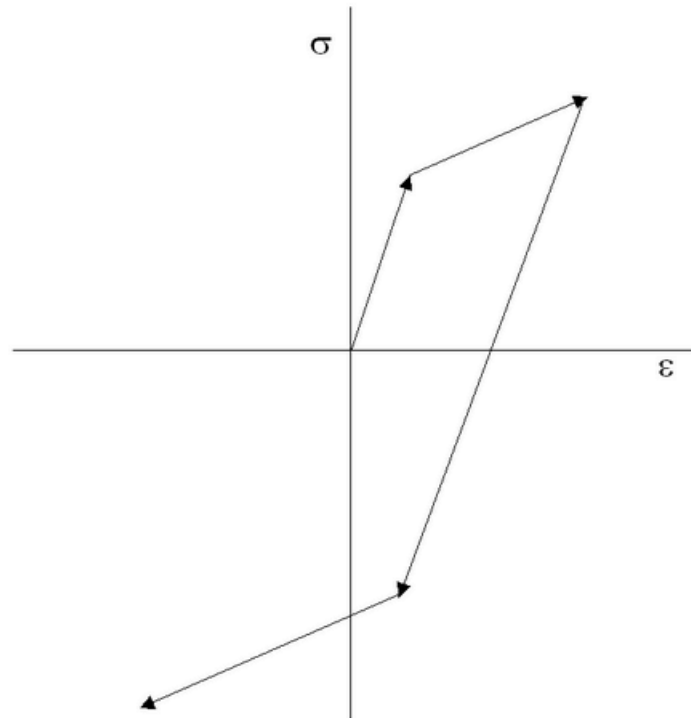
$$\sigma_{VM} = \frac{1}{\sqrt{2}} \left[(\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2 + (\sigma_z - \sigma_x)^2 + 6(\tau_{xy}^2 + \tau_{yz}^2 + \tau_{zx}^2) \right]^{\frac{1}{2}}$$

Yield occurs when

$$\sigma_{VM} = \sigma_{yield}$$

The yield and ultimate stress and the hardening parameters are not used in linear analysis. For non-linear analysis in GSA, only the yield stress is used and ultimate stress and hardening parameters are ignored.

In general, material yielding follows the line defined by the hardening modulus. Either isotropic hardening ($\beta = 1$) or kinematic hardening ($\beta = 0$) can be defined.



If the hardening parameter, β is 1 (isotropic hardening as shown above) the yield stress retains its maximum value on reversal of stress. A value of 0 corresponds to kinematic hardening, where the diameter of the yield surface remains constant so on reversal of stress the material yield when the stress reversal is twice the original yield stress.

Missing Mass & Residual Rigid Response

A modal analysis takes account of how the mass is mobilised in a dynamic analysis. However only a relatively small number of modes are calculated, so not all of the mass is mobilised. Provided the modes up to a high enough frequency are calculated the remaining response can be considered as an essentially static response. There are procedures for establish the missing mass and taking this into account through a static analysis. One is given by the U.S Nuclear Regulatory Commission³.

For a given excitation direction j the mass associated with the modes can be calculated for each degree of freedom from

$$\hat{m}_i = \sum_{n=1}^N \Gamma_{nj} \phi_{ni}$$

³ U.S Nuclear Regulatory Commission, Regulatory Guide 1.92 Combining Modal responses and Spatial Components in Seismic Response Analysis, Revision 2, July 2006

The missing mass is then

$$m_i - \hat{m}_i$$

Given the ground zero period acceleration (ZPA) The missing response can be treated as a static load case

$$f_i = (m_i - \hat{m}_i) a_{ZPA}$$

A set of static loads corresponding to the different directions can then be established/

Gupta Method

The Gupta method is a way of including the residual rigid response along with the response spectrum analysis. This defines a rigid response coefficient, α_i so that the periodic response is

$$r_{pi} = [1 - \alpha_i^2]^{1/2} r_i$$

The coefficient α is defined by Gupta as

$$\alpha_i = \begin{cases} 0 & f_i \leq f_1 \\ \frac{\ln(f_i/f_1)}{\ln(f_2/f_1)} & f_1 \leq f_i \leq f_2 \\ 1 & f_i \geq f_2 \end{cases}$$

where

$$f_1 = \frac{\alpha_{spectral,max}}{2\pi v_{spectral,max}}, \quad f_2 = f_{rigid}$$

Participation Factor and Effective Mass

The modal mass for mode i is defined as

$$\hat{m}_i = \boldsymbol{\varphi}_i^T \mathbf{M} \boldsymbol{\varphi}_i$$

The direction information can be extracted using the participation factor. The participation factor for mode i in the j direction is given by

$$\Gamma_{ij} = \frac{\boldsymbol{\varphi}_i^T \mathbf{M} \mathbf{r}_j}{\hat{m}_i}$$

where the \mathbf{r}_j vector is a rigid body vector in the j direction. The effective mass is similar but defined as

$$\tilde{m}_{ij} = \frac{(\boldsymbol{\varphi}_i^T \mathbf{M} \mathbf{r}_j)^2}{\hat{m}_i} = \hat{m}_i \Gamma_{ij}^2$$

The rigid body vectors are defined as

$$r_x = \left\{ \begin{array}{c} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ \vdots \end{array} \right\}, \quad r_y = \left\{ \begin{array}{c} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ \vdots \end{array} \right\}, \quad \dots$$

So a rigid body vector for a rotation of θ about global z can be defined as

$$r_\theta = \left\{ \begin{array}{c} \cos \theta \\ \sin \theta \\ 0 \\ 0 \\ 0 \\ 0 \\ \cos \theta \\ \sin \theta \\ \vdots \end{array} \right\}$$

The effective mass in a rotated axis system can be calculated from the participation factors and effective masses.

$$\Gamma_{i\theta} = \frac{\varphi_i^T \mathbf{M} \mathbf{r}_\theta}{\hat{m}_i} = \frac{\varphi_i^T \mathbf{M} \mathbf{r}_x}{\hat{m}_i} \cos \theta + \frac{\varphi_i^T \mathbf{M} \mathbf{r}_y}{\hat{m}_i} \sin \theta = \Gamma_{ix} \cos \theta + \Gamma_{iy} \sin \theta$$

So

$$\tilde{m}_{i\theta} = \hat{m} \Gamma_{i\theta}^2 = \hat{m} (\Gamma_{ix} \cos \theta + \Gamma_{iy} \sin \theta)^2$$

The sum of the effective mass in any given direction over all the modes is the total mass. Starting with the definition of effective mass

$$\tilde{m}_{ij} = \frac{(\varphi_i^T \mathbf{M} \mathbf{r}_j)^2}{\hat{m}_i}$$

The rigid body vector can be written as

$$\mathbf{r}_j = \Phi \mathbf{a}_j$$

So the term in the numerator of the effective mass becomes $\varphi_i^T \mathbf{M} \Phi \mathbf{a}_j$ so

$$\tilde{m}_{ij} = \frac{(\varphi_i^T \mathbf{M} \Phi \mathbf{a}_j)^2}{\hat{m}_i} = \frac{(\hat{m}_i a_{ij})^2}{\hat{m}_i} = \hat{m}_i a_{ij}^2$$

Also the total mass $\mathbf{r}_j^T \mathbf{M} \mathbf{r}_j = \mathbf{a}_j^T \Phi^T \mathbf{M} \Phi \mathbf{a}_j$ and $\Phi^T \mathbf{M} \Phi = \text{diag}(\tilde{m}_i)$ so

$$\mathbf{a}_j^T \Phi^T \mathbf{M} \Phi \mathbf{a}_j = \sum_i \tilde{m}_i a_{ij}^2$$

So the sum of the effective masses over all the modes is the total mass.

Patterned Load Analysis

By the principle of superposition for linear elastic structural systems, the internal force in a section can be calculated as

$$f_A = \iint_A I w dx dy$$

where, A is the floor area domain across the x-y plane, I is the influence surface function across

the x-y plane, and W is an un-factored distributed load function varying across the x-y plane.

For the maximum internal force in a section f_A max resulted under a range of distributed load w_{\max} and w_{\min} can be calculated as

$$f_A = \iint_A I [p \cdot w_{\max} + (1-p) \cdot w_{\min}] dx dy$$

where, p is a binary function related to the influence surface I as

$$p = \begin{cases} 1 & I > 0 \\ 0 & I \leq 0 \end{cases}$$

And thus the equation can further be rewritten as

$$f_A = \iint_A I \left(\frac{w_{\max} + w_{\min}}{2} \right) dx dy + \iint_A \left| I \left(\frac{w_{\max} - w_{\min}}{2} \right) \right| dx dy$$

The floor area domain A can always be separated into a series of smaller and non-overlapping area a_i , which exclusively covers the entire area. Assume the sign of I in each individually separated area a_i does not change, i.e. I is always positive or negative across the x-y plane within an area a_i , then the equation can be expanded as

$$f_A = \iint_A I \left(\frac{w_{\max} + w_{\min}}{2} \right) dx dy + \sum_i \iint_{a_i} \left| I \left(\frac{w_{\max} - w_{\min}}{2} \right) \right| dx dy$$

which can be further simplified as an absolute sum function

$$f_{A \max} = f_{mean} + \sum_i |\Delta f_i|$$

where by definition

$$f_{mean} = \frac{1}{2} \iint_A I w_{\max} dx dy + \frac{1}{2} \iint_A I w_{\min} dx dy$$

$$\Delta f_i = \frac{1}{2} \iint_{a_i} I w_{\max} dx dy - \frac{1}{2} \iint_{a_i} I w_{\min} dx dy$$

And similarly, the minimum internal force in a section $f_{A, \min}$ can be derived as

$$f_{A \min} = f_{mean} - \sum_i |\Delta f_i|$$

In most situations, w_{\max} and w_{\min} differ only by a scalar factor, which is related to the load factor of

safety in ultimate limit state design. Putting

$$W_{\min} = S_{\min} W$$

$$W_{\max} = S_{\max} W$$

the equations can be simplified as

$$f_{mean} = \frac{S_{\max} + S_{\min}}{2} \iint_A I w dx dy$$

$$\Delta f_i = \frac{S_{\max} - S_{\min}}{2} \iint_A I w dx dy$$

By comparing these equations to first equation, it can be seen that f_{mean} can be evaluated directly from the analysis with all area fully loaded, and Δf_i can be evaluated directly from the analysis with load being only applied to the area a_i , which means the equations can be further simplified as

$$f_{mean} = \frac{S_{\max} + S_{\min}}{2} f_A$$

$$\Delta f_i = \frac{S_{\max} - S_{\min}}{2} f_A$$

This item was written by Ir. Dr. Don Y.B. Ho of Ove Arup & Partners, Hong Kong Ltd and is reproduced here with permission

Raft Analysis

Raft analysis is a soil-structure interaction analysis, iterating between a solution of the structural analysis using Gss and the soil analysis using Pdisp until convergence of nodal displacements is achieved.

Iteration

For the soil interaction nodes on raft, the analysis iterates through the following steps until convergence is reached:

1. For each soil interaction node, a spring support will be generated with default support stiffness if it does not exist. If the support spring exists, it will be used and its stiffness will be modified during the analysis. After analysis, its stiffness will be restored.
2. Run Gss linear static analysis to obtain the displacements u_{raft} and spring-support forces f_i for each of the interaction nodes
3. Calculate soil pressure p_i under each of the soil interaction nodes using

$$p_i = \frac{f_i}{A_i}$$

$$p_i = \max(p_i, p_{\min})$$

$$p_i = \min(p_i, p_{\max})$$

Where A_i is the area associated with interaction node and the contact pressure is subject to lower and upper limits $[p_{\min} : p_{\max}]$.

4. Run Pdisp analysis to obtain the settlements of soil, u_{soil} under the applied pressure loads
5. Check the differences between raft displacements and soil settlements, if they are smaller than the residual limit, save the results and stop the analysis, otherwise go to step 6
6. Re-calculate the support spring stiffnesses according to the support spring forces and the soil settlements using the following equation and go to step 2.

$$k_i = \frac{f_i}{u_{soil}}$$

A damping coefficient (δ) that can be used to modify the stiffness update, i.e. reserve some percentage of the previous stiffness. The value of damping coefficient is between 0 and 1. If damping coefficient is specified, the new stiffness will be calculated from:

$$k_i = \delta \cdot k_{i,prev} + (1 - \delta) \frac{f_i}{u_{soil}}$$

Piles

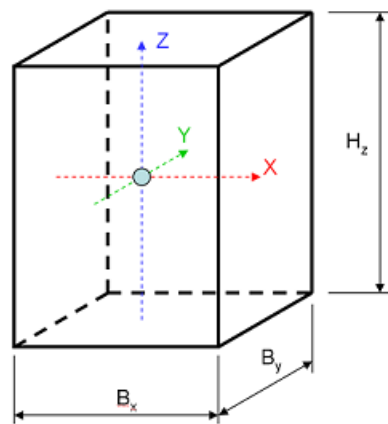
For the soil interaction nodes on the piles, the analysis iterates through the following steps until convergence is reached:

1. For each soil interaction node, generate a spring support in X, Y & Z direction with support stiffness calculated from soil settlement under unit point load.
2. Run Gss linear static analysis to obtain the displacements u_{raft} and spring-support forces f_i for each of the interaction nodes
3. Calculate the soil reaction forces to the interaction nodes from

$$f_x = A_x \sigma_x \quad f_y = A_y \sigma_y \quad f_z = A_z \tau_z$$

where the interaction areas for the interaction node are

$$A_x = B_y H_z \quad A_y = B_x H_z \quad A_z = P H_z$$

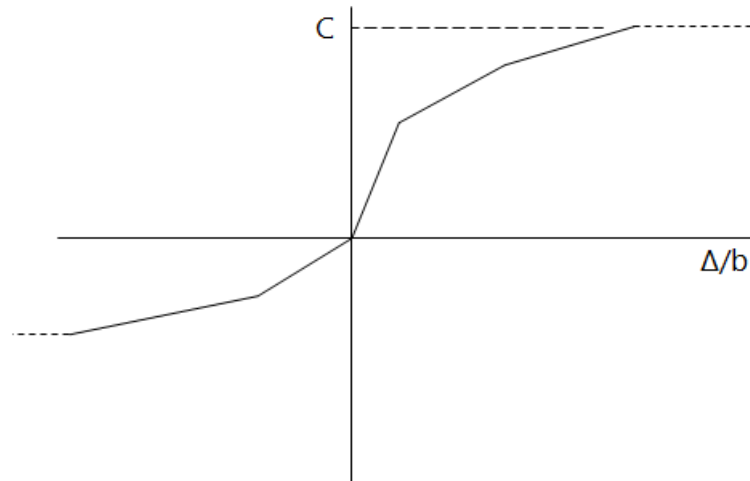


Pile interaction node and relevant dimensions

with P the perimeter of the pile and where

$$\sigma_x = C_x \sigma_{x,max} \quad \sigma_y = C_y \sigma_{y,max} \quad \tau_x = C_z \tau_{z,max}$$

and C_x, C_y, C_z are the pile soil interaction coefficients and derived from the Pile Soil Interaction Coefficient (PSIC) curves defined by the users as shown below.



Pile Soil Interaction Coefficient Curve

where Δ is the differences of the pile displacements and the soil settlements at the corresponding points and b is the pile dimension B_x, B_y in x and y directions or D diameter.

4. Calculate the pressure loads on soil due to the soil reaction forces (f_x, f_y, f_z) and the interaction areas (A_x, A_y, A_z)
5. Run a soil settlement analysis using the embedded Pdisp program
6. Calculate the compensation forces to counter balance the support spring forces due to the use of constant support spring stiffness.
7. Check convergence, if satisfied, stop, otherwise, goto Step 2.

Convergence

The solution is converged if the difference of raft/pile displacements and soil settlements are smaller than the predefined acceptable residual. The residual can be defined in two ways.

- Absolute residual - the residual is defined directly such 0.1 mm, 1.0 mm or 5.0 mm.
- Relative residual - a percentage is defined that is used to calculate the actual residual based on the largest soil settlement. The actual residual is equal to the defined percentage of the largest soil settlement.

Rayleigh Damping

Rayleigh damping considers damping to be related to both the mass and stiffness

$$C = \alpha M + \beta K$$

For a critical damping ratio

$$\xi_i = \frac{1}{2\omega_i}\alpha + \frac{\omega_i}{2}\beta$$

For two distinct damping ratios

$$\begin{Bmatrix} \xi_i \\ \xi_j \end{Bmatrix} = \frac{1}{2} \begin{bmatrix} 1/\omega_i & \omega_i \\ 1/\omega_j & \omega_j \end{bmatrix} \begin{Bmatrix} \alpha \\ \beta \end{Bmatrix}$$

This can be solved to determine the coefficients α, β . In the case of the same damping at both frequencies this simplifies to

$$\beta = \frac{2\xi}{\omega_i + \omega_j}$$

$$\alpha = \omega_i \omega_j \beta$$

RC Slab

Introduction

RCSlab is a design postprocessor within GSA for reinforced concrete two-dimensional elements of uniform thickness subject to any combination of in-plane axial or shear force and out-of-plane bending moment and torsion. The calculations can be made following the principles of most of the codes used within Arup. RCSlab is unable to allow for out-of-plane shear and through-thickness forces.⁴

The input to the postprocessor comprises applied forces and moments, section depth, reinforcement positions, and material properties. The reinforcement orientations can be in general directions, referred to as $\square 1$ and $\square 2$, which need not be orthogonal. The results comprise either areas of reinforcement for each face of the section in the two specified directions, or else an indicator to the effect that RCSlab is unable to find a solution for the current data. Early versions of the program were known as RC2D.

Data Requirements

Each run of *RCSlab* obtains the following data in any consistent set of units from the GSA analysis or *RCSlab* design data as appropriate:

- N_{xx} ultimate applied axial force per unit width in the x-direction
- N_{yy} ultimate applied axial force per unit width in the y-direction

⁴ This item is based on a Notes n Structures written by Ian Feltham [an Arup internal publication] (2007) and is reproduced here with permission

M_{xx}	ultimate applied bending moment per unit width about the x-axis
M_{yy}	ultimate applied bending moment per unit width about the y-axis
N_{xy}	ultimate applied in-plane shear force per unit width
M_{xy}	ultimate applied torsion moment per unit width
e_{add}	additional eccentricity ($e_{add} > 0$) – considered as acting in both senses
e_{min}	minimum eccentricity ($e_{min} > 0$) – considered as acting in both senses
h	section thickness ($h > 0$)
z_{t1}	position of top reinforcement centroid in direction 1 ($0 < z_{t1} < h/2$)
z_{t2}	position of top reinforcement centroid in direction 2 ($0 < z_{t2} < h/2$)
z_{b1}	position of bottom reinforcement centroid in direction 1 ($-h/2 < z_{b1} < 0$)
z_{b2}	position of bottom reinforcement centroid in direction 2 ($-h/2 < z_{b2} < 0$)
θ_1	angle of reinforcement in direction 1, anticlockwise with respect to x-axis
θ_2	angle of reinforcement in direction 2, anticlockwise with respect to x-axis
$A_{st1,min}$	minimum top reinforcement to be provided in direction 1 ($0 < A_{st1,min}$)
$A_{st2,min}$	minimum top reinforcement to be provided in direction 2 ($0 < A_{st2,min}$)
$A_{sb1,min}$	minimum bottom reinforcement to be provided in direction 1 ($0 < A_{sb1,min}$)
$A_{sb2,min}$	minimum bottom reinforcement to be provided in direction 2 ($0 < A_{sb2,min}$)
f_{cd}	compressive design strength of concrete ($f_{cd} > 0$)
$f_{cd,t}$	compressive design strength of top layer of concrete ($f_{cd} > 0$)
$f_{cd,b}$	compressive design strength of bottom layer of concrete ($f_{cd} > 0$)
f_{cdc}	cracked compressive design strength of concrete ($f_{cdc} > 0$)
f_{cdu}	uncracked compressive design strength of concrete ($f_{cdu} > 0$)
f_{cdt}	tensile design strength of concrete ($f_{cdt} > 0$)
ϵ_{ctrans}	compressive plateau concrete strain ($\epsilon_{ctrans} \geq 0$)
ϵ_{cax}	maximum axial compressive concrete strain ($\epsilon_{cax} \geq \epsilon_{ctrans}$)
ϵ_{cu}	maximum flexural compressive concrete strain ($\epsilon_{cu} \geq \epsilon_{cax}$)
β	proportion of depth to neutral axis over which rectangular stress block acts ($\beta \leq 1$)
$(x/d)_{max}$	maximum value of x/d , the ratio of neutral axis to effective depth, for flexure: $(x/d)_{min} < (x/d)_{max} \leq 0.5 / [0.5 + \min\{z_{t1}, z_{t2}, -z_{b1}, -z_{b2}\} / h]$
E_s	elastic modulus of reinforcement

f_{yd}	design strength of reinforcement in tension ($f_{yd} > 0$)
f_{ydc}	design strength of reinforcement in compression, ($f_{ydc} > 0$)
f_{lim}	maximum linear steel stress of reinforcement ($f_{lim} > 0$)
ϵ_{plas}	yield strain of reinforcement in tension ($\epsilon_{plas} > 0$)
ϵ_{plasc}	yield strain of reinforcement in compression ($\epsilon_{plasc} > 0$)
ϵ_{su}	design value of maximum strain in reinforcement
φ_{Δ}	maximum permitted angle between applied and resulting principal stress

In addition, the program needs to know whether to use, where appropriate, the faster approach and, if so, what the maximum area of reinforcement so calculated should be before the rigorous approach is used.

Within *RCSlab* the reinforcement positions are measured with respect to the mid-height of the section, the positions being measured positively upwards. The reinforcement angles are specified with respect to the x-axis and measured positively in an anticlockwise direction looking from above. It should be noted that the concrete is assumed to have zero tensile strength in the analysis; the tensile strength, f_{ctr} , is only used to calculate the compressive strength when tensile strains are present.

The results of each run consist of the required area of reinforcement, negative if tensile, in each direction in the top and bottom faces or an error flag indicating that a solution could not be found.

Other Symbols Used in Theory

stress

α	proportion of total thickness comprising each outer layer
ϵ	strain
φ	angle of principal compressive stress and strain, measured anticlockwise with respect to x axis

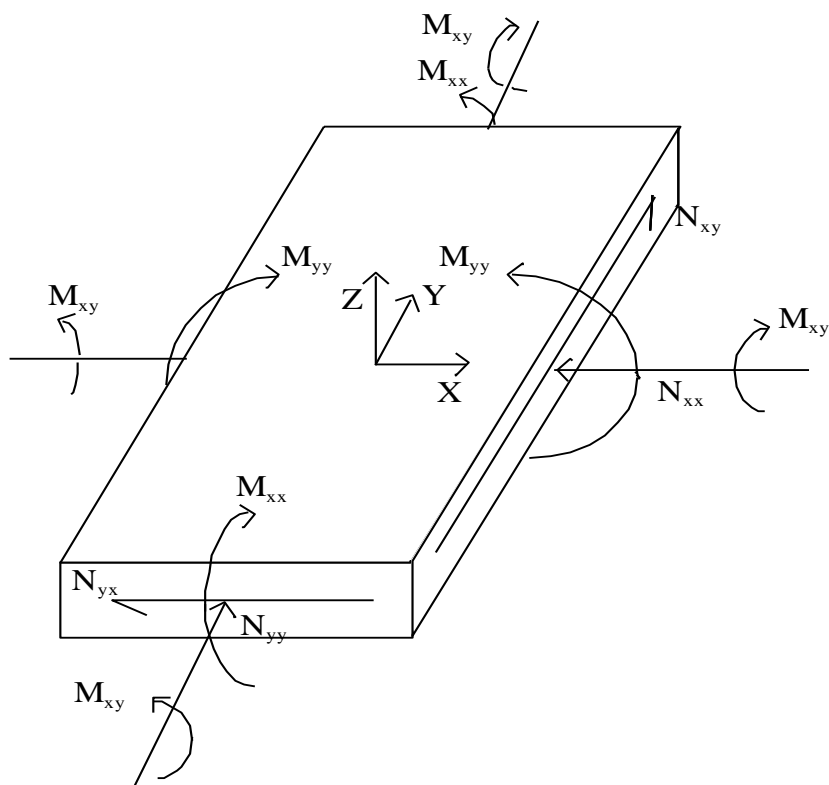
Suffices

A	relating to pass A
B	relating to pass B
C	relating to pass C or compression-only reinforcement
T	relating to tension-only reinforcement
a	applied
b	bottom layer or face
c	concrete

- e effective
- m middle layer
- n general direction = 1 or 2
- s steel
- t top layer or face
- z at level of reinforcement
- 1 direction 1
- 2 direction 2
- > principal direction with greater compression or lesser tension
- < principal direction with greater tension or lesser compression

RC Slab Sign Convention

Within RClab the sign convention is as shown below; note that although it is different from that used by GSA, the interface converts as necessary:



Compression and sagging moment are positive.

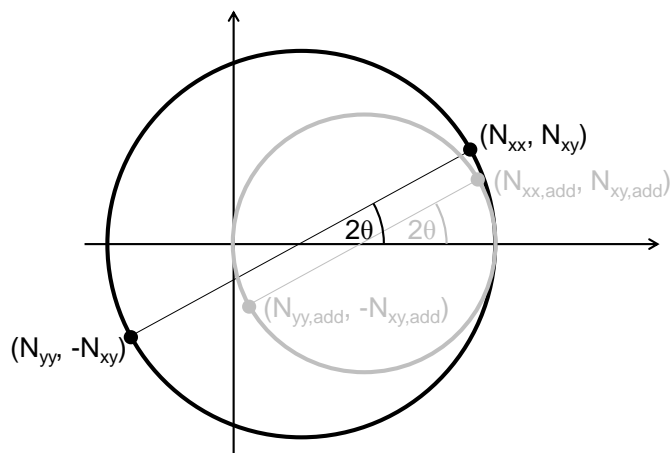
RC Slab Analysis Procedure

The following summarizes the procedure followed by *RCSlab*:

1. Adjust, where necessary, the applied moments for minimum eccentricities.
2. Split the section into three layers with the central layer unstressed and the outer layers taking in-plane stresses, the thicknesses corresponding to an acceptable neutral axis depth; calculate the stresses applied to each layer.
3. Calculate the stress to be taken by the concrete in each layer and the stress from each layer to be taken by reinforcement.
4. Calculate the force to be taken by each of the four sets of reinforcement (two faces, two directions) taking into account their positions relative to the layers.
5. Determine section strains compatible with the neutral axis depths implied by the layer thicknesses in 5.0.2 and the concrete strains in the outer layers from 5.0.3 for top and bottom layers.
6. Determine reinforcement strains compatible with the section strains.
7. From the strains, calculate the stress in each of the four sets of reinforcement.
8. Knowing the force to be taken by each of the four sets of reinforcement and the stress in each set of reinforcement, calculate the reinforcement areas required; these should not be less than the specified minimum values.
9. Repeat as necessary from 5.0.2, adjusting the layer thicknesses to achieve the minimum total area of reinforcement.
10. Where in-plane effects dominate, repeat from 5.0.2 adopting a model with the central layer stressed.
11. The design reinforcement areas correspond to the layer arrangement giving the minimum total area of reinforcement.
12. To speed up the calculation, an option is available to adopt a non-iterative technique where the loading is primarily either in-plane or out-of-plane.

Inclusion of moments resulting from additional and minimum eccentricities

The applied moments are adjusted to take into account the additional and minimum eccentricities of applied axial forces. The additional eccentricity, which can be used to model tolerances and second-order effects, is determined by the user; applied bending moments are increased by compressive principal axial forces but are not adjusted for tensile principal axial forces. The components of in-plane force in the orthogonal directions for use with the additional eccentricity, $N_{xx,add}$, $N_{yy,add}$ and $N_{xy,add}$, are calculated assuming the angle between the principal direction and the x-axis is unchanged.



The default value of the minimum eccentricity, which can be overwritten, is taken from the chosen design code; this value, and all other code-dependent values, are given in Appendix 3. If the absolute value of the applied moment exceeds the sum of the additional and minimum eccentricity moments for M_{xx} , M_{yy} and M_{xy} , then the applied moments are increased in magnitude by their respective additional moments. Otherwise two sets of applied moments are calculated, corresponding to eccentricities applied in the two senses.

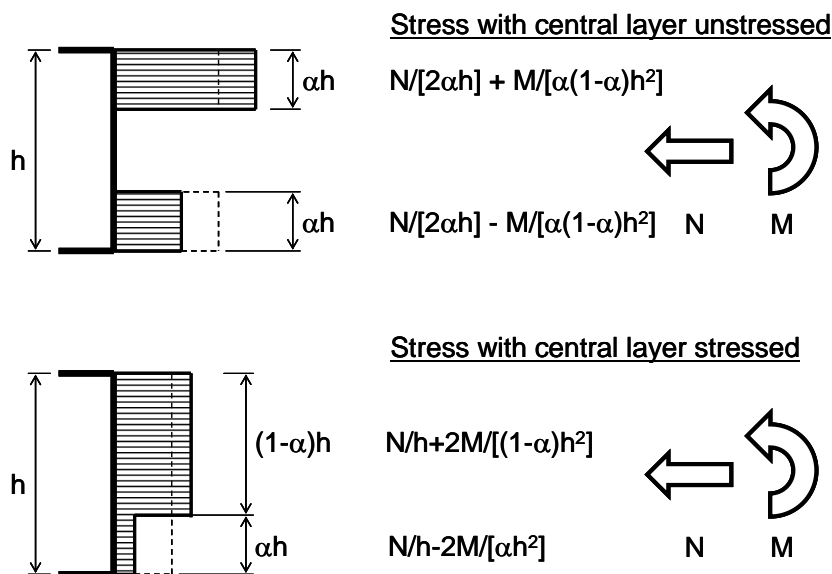
$$M_d = \max \left\{ \begin{array}{l} \min \left\{ \begin{array}{l} \max \left\{ \begin{array}{l} M + M_{add} \\ M_{min} \end{array} \right\} \\ M + j \times (M_{add} + M_{min}) \end{array} \right\} \\ \min \left\{ \begin{array}{l} M - M_{add} \\ -M_{min} \end{array} \right\} \end{array} \right\}$$

where j takes the value of -1 for the first set and +1 for the second set.

For example, if the applied, additional and minimum eccentricity moments were 75kNm, 50kNm and 60kNm respectively, the design moments for the two sets would be 75-50-60 = -35kNm and 75+50 = 125kNm respectively. It should be noted that no specific allowance is made for slenderness.

Division of section into layers

The section is divided into three layers, each with constant in-plane stress. The outer layers both have a thickness of αh , where α is the proportion of the total section thickness, h ; the inner layer is therefore of thickness $(1-2\alpha)h$. In-plane forces are resisted either by the outer layers alone or, if there is sufficient in-plane compression, by all three layers. The outer layers also resist the moments resulting from out-of-plane forces.



(In this figure, N represents N_{xx} , N_{yy} or N_{xy} and M represents M_{yy} , M_{xx} or M_{xy} respectively.)

The program determines iteratively the value of α that gives the lowest total area of reinforcement, α_{opt} . Calculations are first performed for the condition with the central layer unstressed, and then repeated for the condition with it stressed, although the latter results are only valid if there is no tension reinforcement, or if tension reinforcement is required in both faces, in a particular direction.

The minimum valid value for α is zero; any code-specified maximum lever arm is converted into an equivalent maximum strain in the reinforcement, which is explicitly checked. Where the central layer is unstressed, and the axial compression in the reinforcement direction with the larger force eccentricity is less than $0.1f_{cd,u}.h$, the maximum valid value α is given by $\alpha_{max} = \beta(x/d)_{max} \cdot (0.5 + \min\{z_{t1}, z_{t2}, -z_{b1}, -z_{b2}\}/h)$; where the axial compression in this direction exceeds $0.5\beta f_{cd,u}.h$, $\alpha\beta_{max} = 0.5$; for intermediate values of axial compression, the value of α_{max} is linearly interpolated between these two extremes. Where all three layers are stressed, the maximum valid value for α is given by $\alpha_{max} = 0.5$.

The initial value of α is taken as $\alpha_{max}/2$. If this value does not give a valid result, calculations are performed for $\alpha = \alpha_{max}/4$ and $\alpha = 3\alpha_{max}/4$. If neither of these values results in a valid result, the additional four values at the eighth points of the range are calculated. This process is repeated, using sixteenth points etc, until there is at least one valid result or the interval between adjacent values of α falls below a limiting value of 0.02.

Once a valid value of α has been determined, only the values lying between this value and the previously calculated values of α on either side are considered, halving the interval. This is repeated, halving the interval for each turn, until it falls below a limiting value of 0.005.

In the following example, it has been assumed that $\alpha_{min} = 0$, $\alpha_{max} = 0.5$, valid results are obtained for $0.40 \leq \alpha \leq 0.45$, $\alpha_{opt} = 0.4321$, and, for simplicity, the area of reinforcement increases linearly with the difference between α and α_{opt} . The optimisation process stops when the interval is less than 0.005. The program would calculate areas of reinforcement for the following values of α .

Values of α in *italics* indicate no valid solution; values in **bold** indicate this is a revised optimum value.

Values of α for which reinforcement areas calculated				Current value of α_{opt}	Current value of interval
0.25				-	0.25
0.125		0.375		-	0.125
0.0625	0.1875	0.3125	0.4375	0.4375	0.0625
0.40625		0.46875		0.4375	0.03125
0.421875		0.453125		0.4375	0.015625
0.4296875		0.4453125		0.4296875	0.0078125

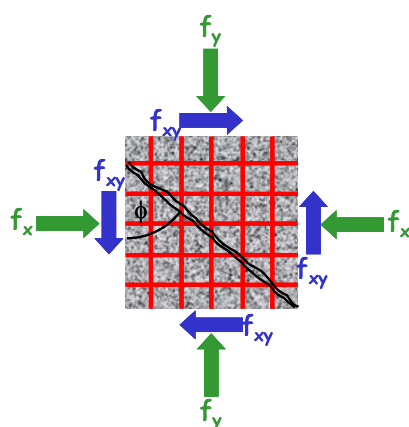
The following procedure must be completed for each trial value of α

Calculation of stress to be taken by reinforcement in a layer

The design strength of the concrete, f_{cd} , depends on the extent of cracking and needs not be the same in different layers. For each layer, the principal applied stresses, $f_{a<}$ and $f_{a>}$ with $f_{a<}$ being more tensile, are calculated from the orthogonal stresses. If $f_{a<}$ is compressive, the uncracked concrete strength, f_{cdu} , is used. Otherwise the design strength depends on the ratio of $f_{a<}$ to the tensile strength of the concrete, f_{ctd} ; the strength is calculated as $f_{cdu} \times [1 - (f_{a<}/f_{ctd})^2]$, but not less than the cracked concrete strength, f_{cdc} . The concrete strengths are code dependent.

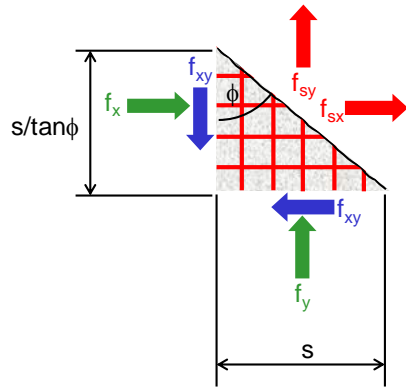
If $f_{a<}$ is not tensile and $f_{a>}$ is not greater than f_{cd} , then reinforcement is not required to take any stress from this layer. Otherwise, the reinforcement must take sufficient stress to achieve equilibrium without either overstressing the concrete or requiring it to take tension. The general expressions are given in Appendix 1.

The following diagrams illustrate the situation within a layer where the reinforcement is aligned with the x and y directions.



Consider a 2D element of uniform thickness with applied in-plane normal stresses f_x and f_y and shear stress f_{xy} . Assume that the section cracks in a plane at angle ϕ to the x plane, measured anticlockwise. The applied forces are resisted by the concrete, acting in compression, and reinforcement in the orthogonal directions taking tension or compression as necessary.

The angle at which the cracks form under the design loads, ϕ , is not generally equal to the angle of the initial cracks, which depends on the direction of principal tensile stress. As reinforcement yields, the angle of cracking rotates, and this angle may be limited by a design code or the user. *RCSlab* selects the angle, complying with any limitation, which results in the lowest total area of reinforcement. This often corresponds to the angle that also gives rise to the minimum stress in the concrete.



Consider equilibrium of the triangle below the crack; f_{sx} and f_{sy} are the stresses taken by reinforcement:

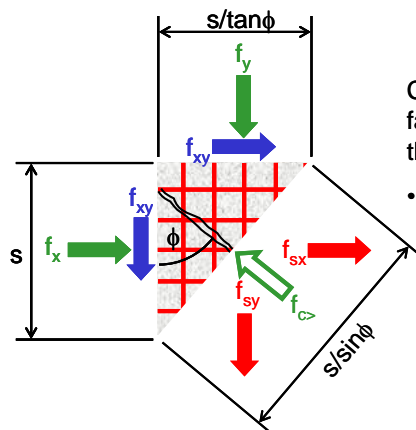
- Resolving horizontally

$$s/\tan\phi \cdot (f_x + f_{sx}) = s \cdot f_{xy}$$

$$\therefore f_{sx} = f_{xy} \cdot \tan\phi - f_x$$
- Resolving vertically

$$s \cdot (f_y + f_{sy}) = s/\tan\phi \cdot f_{xy}$$

$$\therefore f_{sy} = f_{xy}/\tan\phi - f_y$$



Consider equilibrium of the triangle with a face normal to the crack; $f_{c>}$ is the stress in the concrete:

- Resolving horizontally

$$s \cdot (f_x + f_{sx}) + s/\tan\phi \cdot f_{xy} = (s/\sin\phi \cdot f_{c>}) \cdot \sin\phi$$

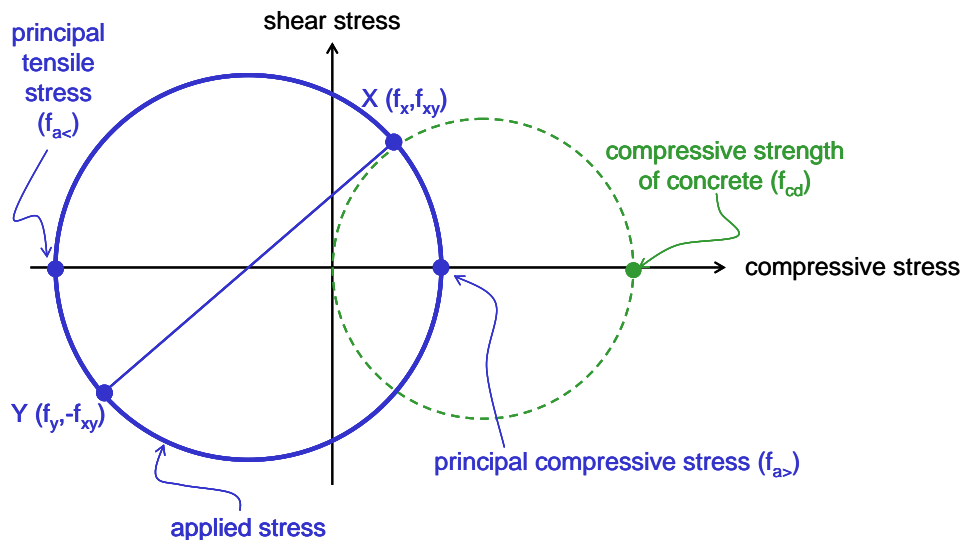
$$\therefore f_{c>} = f_x + f_{sx} + f_{xy}/\tan\phi$$

$$= f_{xy} \cdot \tan\phi + f_{xy}/\tan\phi$$

$$f_{c>} = 2f_{xy}/\sin 2\phi$$

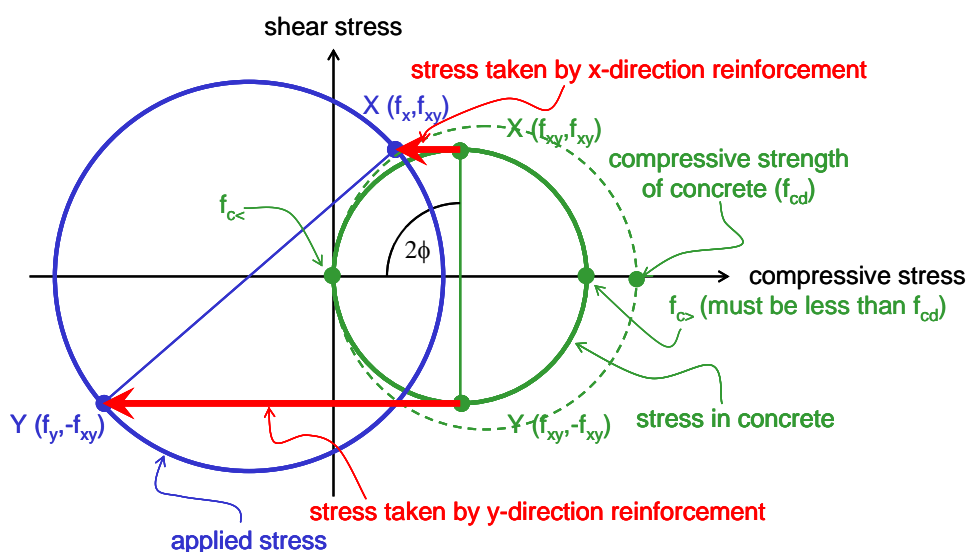
If both f_{sx} and f_{sy} are positive, tensile stress must be taken by reinforcement in both x and y directions. The lowest total stress is required when ϕ equals 45° , for which angle $f_{sx} = f_{xy} - f_x$, $f_{sy} = f_{xy} - f_y$ and $f_{c>} = 2f_{xy}$.

Mohr's circles provide a useful tool for determining the stress to be taken by the reinforcement and the maximum stress in the concrete. The circle representing the compressive strength of concrete has principal values of the origin and the concrete design strength. If **this circle** does not encompass **the circle** representing the applied stress, reinforcement is required.



Reinforcement is required because the applied stress circle does not lie within the circle representing the strength of cracked concrete

The **stress taken by the reinforcement** is subtracted from the **applied stress** to give a residual **stress in the concrete**, which must always fall within the circle representing its compressive strength. If this is not achievable, then it is not possible to reinforce for the applied stress with reinforcement in the directions chosen; the reinforcement must be orientated closer to the principal directions, or else stronger concrete should be used. If the reinforcement is aligned with the x and y axes, then the offsets representing the stress taken by the reinforcement are horizontal; otherwise the offsets are at an angle with the vertical offsets equal in magnitude but of opposite sign.



Note that for $\phi=45^\circ$, the stress taken by x-direction reinforcement is equal to $(f_{xy} - f_x)$ and that taken by the y-direction reinforcement is equal to $(f_{xy} - f_y)$

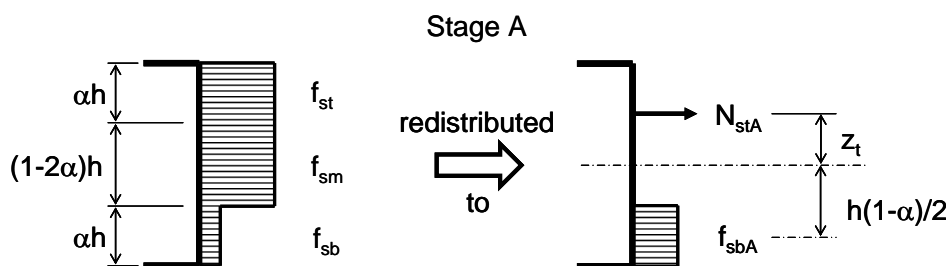
Calculation of forces to be taken by reinforcement

Having determined the stresses to be taken by reinforcement in all three layers, the forces in the four sets of reinforcement must be determined considering equilibrium. The differences between the centroids of the stressed layers and the positions of the reinforcement must be taken into account.

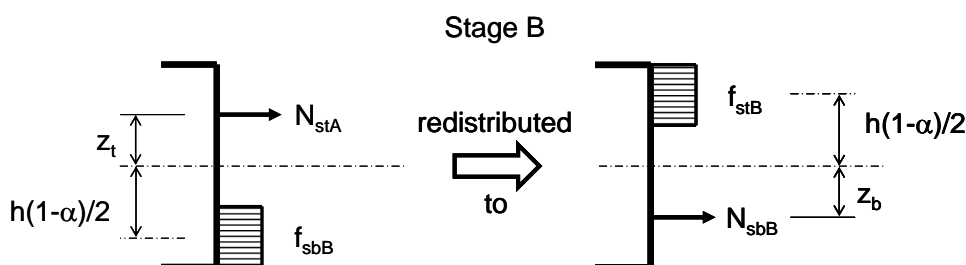
If this calculation is undertaken in one stage, anomalies could occur such as tension reinforcement being required where the surrounding concrete is in biaxial compression. To avoid this, the calculation is made in three stages.

The procedure works better if the larger moment results in tension on the bottom. If the larger moment results in tension at the top, the section and applied forces are inverted before the calculation is undertaken, and the results inverted at the end.

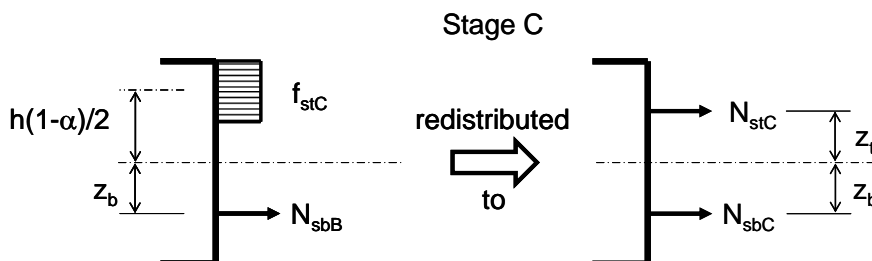
In stage A, the forces required in the top steel, N_{stA} , are calculated assuming that the stresses to be taken by the bottom reinforcement are distributed uniformly across the bottom layer, f_{sbA} . This is shown diagrammatically below; there would of course be an equivalent diagram for the second steel direction. The overall distribution of concrete and steel stresses in the bottom layer is then rechecked to see whether any of f_{sbA} can in fact be resisted by the concrete; this calculation results in stresses of f_{sbB} to be taken by the bottom reinforcement.



In stage B, the forces required in the bottom steel, N_{sbB} , are calculated returning the top reinforcement forces to stresses distributed uniformly across the top layer, f_{stB} . The overall distribution of concrete and steel stresses in the top layer is then rechecked to see whether any of f_{stB} can be resisted by the concrete; this calculation results in stresses of f_{stC} to be taken by the top reinforcement.



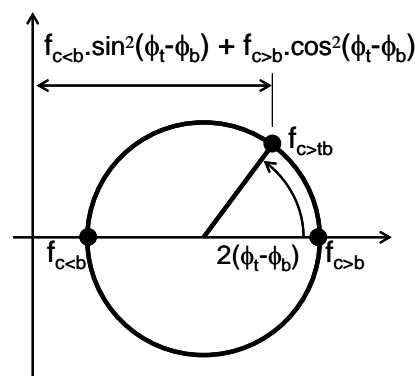
In the final stage, C, any reinforcement, top and bottom, required to take the modified stresses in the top layer is calculated. By considering the top layer last, there will be places where no top steel is required. Although stage C could result in bottom steel in locations where it would be unnecessary with further rationalisation, the reinforcement areas would be small and usually less than nominal bottom steel.



Although this three stage approach calculates reinforcement forces reasonably economically, it may result in small differences in the areas of top and bottom reinforcement for pure axial load.

Determination of section strains compatible with concrete strains in outer layers

To determine the four sets of reinforcement areas required to take these sets of forces, a design strain, compatible with the concrete stress blocks, is needed for each set. The first stage of calculating these strains is to determine the compressive strains at the outer or inner boundaries of the concrete stress blocks. The calculation of the principal compressive strains at the top is described below; a corresponding procedure is used to estimate the principal compressive strains at the bottom.



Calculation of stress $f_{c>tb}$ in bottom layer at the angle of principal strain in the top layer

Although at this point in the procedure, principal compressive stresses in the concrete have been determined for the top, $f_{c>t}$, and bottom, $f_{c>b}$, layers, the directions of these two principal axes, ϕ_t and ϕ_b , will not generally be aligned. In order to estimate the strain profile for the top layer, $f_{c>tb}$, the stress in the bottom layer in the same direction as $f_{c>t}$, is calculated.

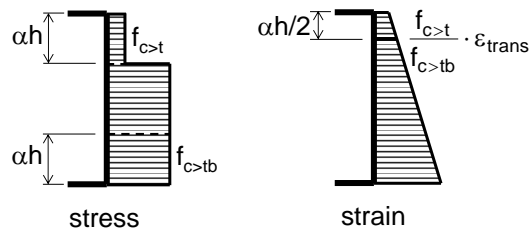
Knowing $f_{c>tb}$ enables the compressive principal strains in the top layer to be calculated.

If $f_{c>t}/f_{cd,t}$ equals $f_{c>tb}/f_{cd,b}$, then the more compressive strain, $\epsilon_{c>t}$, is at the top of the section and taken as the axial limiting strain, ϵ_{cax} .

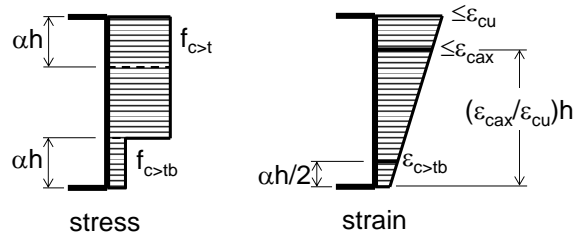
stress

strain

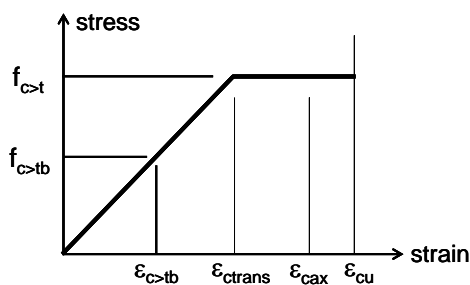
If $f_{c>t}/f_{cd,t}$ is less than $f_{c>tb}/f_{cd,b}$, $\eta_{c>t}$ is at the centre of the top block and calculated as the compressive plateau strain of concrete, ϵ_{ctrans} , factored by $f_{c>t}/f_{c>tb}$.



If $f_{c>t}/f_{cd,t}$ is more than $f_{c>tb}/f_{cd,b}$, $\eta_{c>t}$ is at the top of the section and calculated so that it does not exceed the limiting strain, ϵ_{cu} , and that the axial limiting strain, ϵ_{cax} , is not exceeded at a position $(1-\epsilon_{cax}/\epsilon_{cu})h$ from the face, assuming a strain at the centre of the bottom block of ϵ_{ctrans} factored by $f_{c>tb}/f_{c>t}$.



Assumed stress-strain relation for concrete



The maximum compressive strain on the more compressed face can exceed ϵ_{ctrans} even if the maximum compressive stress is less than the concrete strength f_{cd} since the ultimate strain could be obtained with a smaller stress block, αh , but more favourable lever arm

For all conditions, the less compressive principal strain, $\eta_{c<t}$, is taken at the same level as $\eta_{c>t}$. If the less compressive principal stress, $f_{c<t}$, is compressive, then $\eta_{c<t}$ is taken as ϵ_{ctrans} factored by $f_{c<t}/f_{c>t}$, where $f_{c<t}$ is the less compressive principal stress in the top block. The key concrete strains are code-dependent.

Determination of compatible reinforcement strains

Having estimated the concrete strains, the next stage is to determine compatible reinforcement strains. There are three different procedures for calculating the reinforcement strains, the choice of which depends on the relations between the reinforcement forces, summarized in the following table. Different procedures may be appropriate for each face and direction.

For directions with flexure, the neutral axis is taken as ah/η from the compression edge, where η is the ratio of the depth of the rectangular stress block to the depth of the neutral axis, which is code-dependent.

C = compression N = none T = tension U = none; concrete unstressed				Procedure	Principles in determining reinforcement strain
Face considered		Opposite face			
Direction considered	Other direction	Direction considered	Other direction		
C T	T C	Any T/U	T/U Any	A	Principal tensile strain chosen to give minimum area of reinforcement; calculations made at level of compression reinforcement.
Of the remainder:					
T T	T/U C/N	T/U T/U	T/U C/N	B	Strain compatibility not required since both faces in tension and no neutral axis in other direction.
All others				C	If concrete cracked, boundary strain calculated to be compatible with principal compressive strain on opposite face and neutral axis depth consistent with depth of stress block. Strain at reinforcement level interpolated from boundary strains.

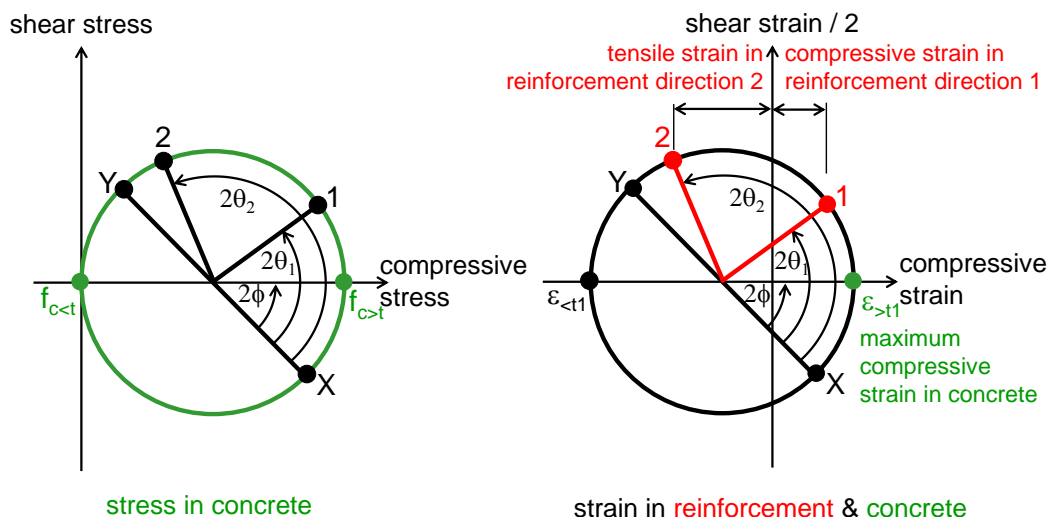
The procedures are described below in detail.

Procedure A

The principal tensile strain is calculated so that the total area of reinforcement in this layer is at a minimum. Note that if the tensile strain is not large enough, the tensile reinforcement will be working at a low stress, or even be in compression, but if the tensile strain is too large, the compressive reinforcement will be working at a low stress, or even be in tension.

The formulae to calculate the optimum value of the tensile strain are given in Appendix 2. The calculation is made at the level of the compression reinforcement. Although the tension reinforcement will be at a slightly different level, it is assumed that the strain in it is approximately equal to the strain in its direction at this level; since the reinforcement in the opposite face in the same direction will always be in tension, this should be a reasonable assumption.

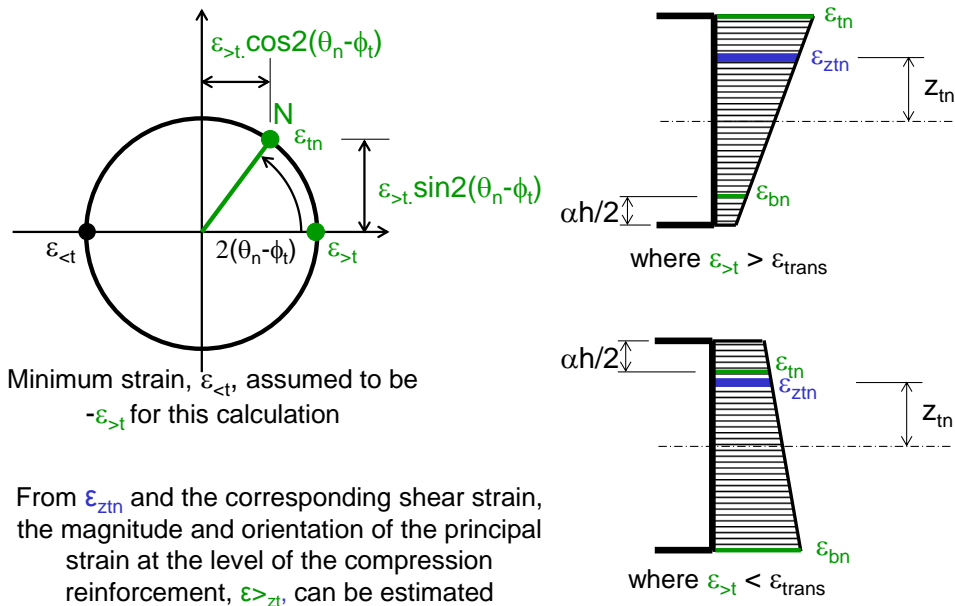
Relations between stresses and strains at the level of the top reinforcement, direction 1, with procedure A



Note that principal stresses and strains are aligned
 The notation assumes that this is the top layer with compression reinforcement in direction 1

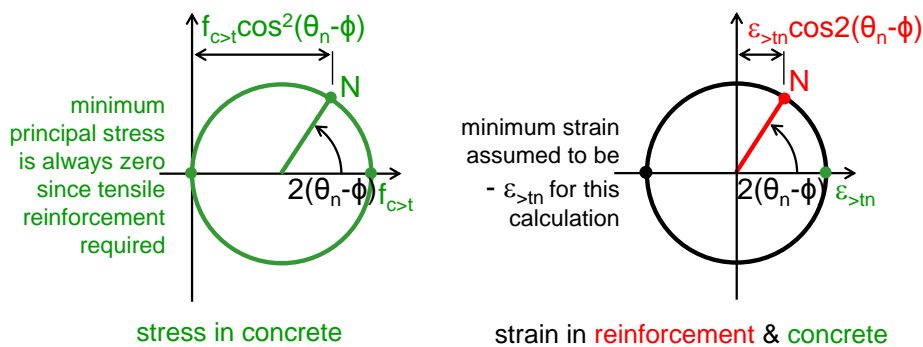
To estimate the principal compressive strain at the level of the compression reinforcement, the variation in magnitude and direction of the principal compression between the top and bottom layers must be considered. For these preliminary calculations, the magnitude of the principal tensile strain is taken as equal to that of the principal compressive strain. The strains in the reinforcement directions are calculated using the procedure in the figure below for both top and bottom layers at their appropriate block boundary; from these strains, the value at the level of the compression reinforcement is linearly interpolated. From the normal and the shear strains at the level of the compression reinforcement, the magnitude and orientation of the principal strain at this level can be estimated.

Estimating magnitude and orientation of principal strain in concrete at level of compression reinforcement with procedure A



The final stage of calculating reinforcement stresses makes allowance for the area of concrete displaced by compression bars by subtracting the concrete stress at the level of the reinforcement from the steel stress. To compensate for this when determining the optimum tensile strain with procedure A, a force, estimated to equal the force in displaced concrete, is added to compressions. This is achieved by factoring the compression by:

$$1 + \frac{f_{c>t} \cos^2(\theta_n - \phi)}{\min(f_s, E_s \epsilon_{>_{tn}} \cos 2(\theta_n - \phi))}$$



Procedure B

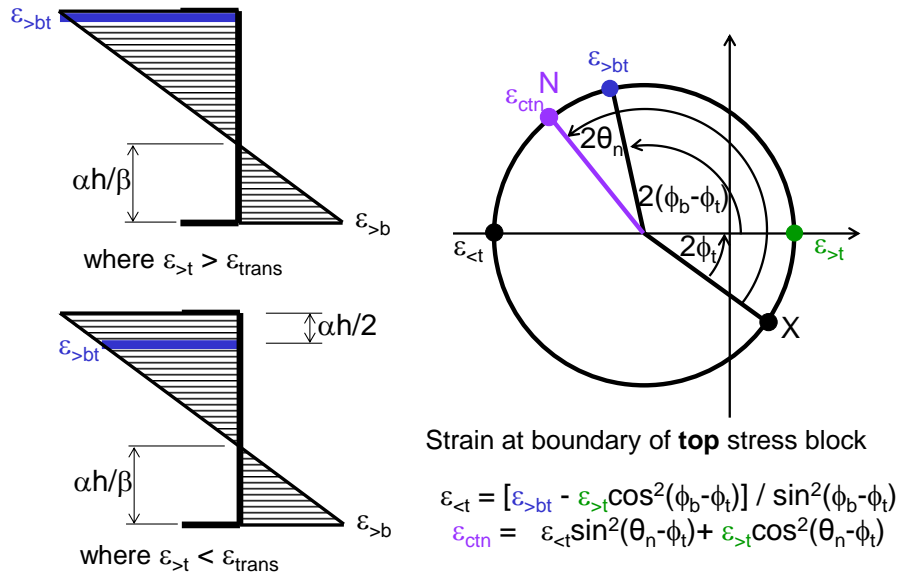
This procedure is adopted when strain compatibility is not required for tension reinforcement; the full design strength of the reinforcement is used.

Procedure C

The first stage of this procedure is to calculate the tensile strain in the compression block at either its surface or inner boundary, as appropriate, in the directions of the reinforcement. If the

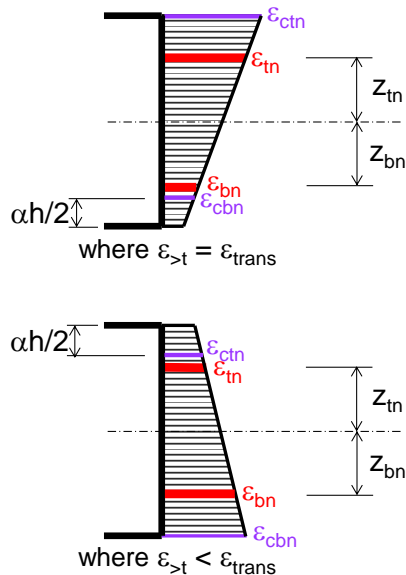
stress block is in biaxial compression, the strain is determined from the concrete stress, but if it is cracked, the strain compatible with the principal compressive strain on the opposite face and the depth of the neutral axis is determined.

Calculating strain in reinforcement directions, ϵ_{ctn} , for cracked top stress block with procedure C



Having obtained the strains in the reinforcement directions at the boundaries of the stress blocks, the strains at the levels of the reinforcement can be calculated.

Calculating reinforcement strains, ϵ_{tn} and ϵ_{bn} , with procedure C



Determination of stress in reinforcement

The stress in the reinforcement is calculated from the strain; a tri-linear stress-strain relation is used for the reinforcing steel in both tension and compression, for which the salient points are

code-dependent. If the bar is to take compression, the calculated stress in the reinforcement is reduced by the layer's concrete stress in the reinforcement direction to allow for the displaced concrete.

Calculation of area of reinforcement

If the force and stress are of opposite signs then the calculation for this particular value of α is unsuccessful. If the force required to be taken by the reinforcement is of the same sign as the stress calculated in that steel, the area is calculated by dividing the forces by the stress; the area is limited to half the section thickness. The specified minimum area is used as appropriate. Finally, the four reinforcement areas are summed to give a total that is compared with the totals resulting from other iterations: the reinforcement areas with the smallest total are output by the program, with compressive reinforcement identified by negating its area, following GSA's sign convention.

Non-iterative technique

To speed up the calculation, an option is available, where the loading is primarily either in-plane or out-of-plane, to adopt a non-iterative technique. This approach is likely to lead to slightly more conservative results. The user can choose to use this approach in appropriate situations and can specify a total area of reinforcement, as a percentage of the cross-sectional area, above which a rigorous, iterative solution is used.

Primarily in-plane

The applied loads are taken as primarily in-plane if the eccentricities of the moments equal the minimum eccentricity for each component for which there is a corresponding force.

Where loads are primarily in-plane, α is taken as 0.5. If the maximum compressive stress in the layers exceeds its strength, the section is unreinforceable; otherwise, reinforcement areas are calculated.

Primarily out-of-plane

The applied loads are taken as primarily out-of-plane if the more compressive principal is less than $0.1f_{cd,u}$. Tensile normal stresses do not prevent use of this technique.

Where loads are primarily out-of-plane, the central layer is unstressed and the value of α is initially set to α_{lim} . If the maximum compressive stress in the compressive outer layers exceeds the strength, the section is unreinforceable. If compression reinforcement is required, the iterative solution is used. Otherwise, a reduced value of α is calculated so that the relatively more highly stressed of the top and bottom layers is just below its compressive strength, while remaining in equilibrium with the applied moments; limits on the change in α are imposed. The stresses are recalculated for this value of α , and a further revised value of α calculated. If the change in α exceeds 0.01, the stresses are recalculated again. Reinforcement areas are calculated for the final value of α .

Distribution of reinforcement

RCSlab calculates the area of reinforcement required at each node. Since the reinforcement distribution corresponds to the force and moment distributions with their concentrations and peaks, there may be locations where no satisfactory reinforcement arrangement can be

determined because the concrete is overstressed in shear. If these points, which are left black when contouring, are isolated, they can probably be ignored but larger areas will require changes to the geometry or material properties.

It is also usually appropriate to average values of reinforcement in areas of great change. For example, reinforcement requirements in flat slabs can be averaged over the central half of the column strips, the outer portions of the column strips and the middle strips, as when following code methods. It is hoped that future developments within *GSA* will help automate this averaging process.

Reduced stiffness & P-delta

The element stiffness can be partitioned into structure (s) and retained (r) degrees of freedom

$$\begin{Bmatrix} \mathbf{f}_s \\ \mathbf{f}_r \end{Bmatrix} = \begin{bmatrix} \mathbf{K}_{ss} & \mathbf{K}_{sr} \\ \mathbf{K}_{rs} & \mathbf{K}_{rr} \end{bmatrix} \begin{Bmatrix} \mathbf{u}_s \\ \mathbf{u}_r \end{Bmatrix}$$

So

$$\mathbf{u}_r = \mathbf{K}_{rr}^{-1} \mathbf{f}_r - \mathbf{K}_{rr}^{-1} \mathbf{K}_{rs} \mathbf{u}_s$$

Giving the reduced equation

$$\left(\mathbf{f}_s - \mathbf{K}_{sr} \mathbf{K}_{rr}^{-1} \mathbf{f}_r \right) = \left(\mathbf{K}_{ss} - \mathbf{K}_{sr} \mathbf{K}_{rr}^{-1} \mathbf{K}_{rs} \right) \mathbf{u}_s$$

or

$$\tilde{\mathbf{f}}_s = \tilde{\mathbf{K}}_{ss} \mathbf{u}_s \left[\tilde{\mathbf{K}}_{ss} \right]$$

When creating the structure stiffness matrix the element matrix can be assembled and then reduced as above before being included in the structure equations.

$$\mathbf{f}_s = \mathbf{K}_{ss} \mathbf{u}_s$$

Once the structure displacements are calculated the element displacements can be established from

$$\mathbf{u}_e = \begin{Bmatrix} \mathbf{u}_s \\ \mathbf{K}_{rr}^{-1} \mathbf{f}_r - \mathbf{K}_{rr}^{-1} \mathbf{K}_{rs} \mathbf{u}_s \end{Bmatrix}$$

And the element forces as

$$\mathbf{f}_e = \mathbf{K}_{ee} \mathbf{u}_e$$

For a P-delta analysis the global solution is modified to

$$\mathbf{f}_s = \left(\mathbf{K}_{ss} + \mathbf{K}_{gss} \right) \mathbf{u}_s = \hat{\mathbf{K}}_{ss} \mathbf{u}_s$$

but the element force calculation is unchanged. This means that once the structure displacements are calculated the element displacements and forces are calculated from

$$\mathbf{u}_e = \begin{Bmatrix} \mathbf{u}_s \\ \hat{\mathbf{K}}_{rr}^{-1} \mathbf{f}_r - \hat{\mathbf{K}}_{rr}^{-1} \hat{\mathbf{K}}_{rs} \mathbf{u}_s \end{Bmatrix}$$

$$\mathbf{f}_e = \mathbf{K}_{ee} \mathbf{u}_e$$

Seismic Calculation

Response Spectrum Analysis

Response Spectrum

The response of a single degree of freedom system mode (frequency f , spectral acceleration $a_{spectral}$) is

$$q_i = \frac{a_{spectral}}{(2\pi f)^2}$$

Modal analysis reduces a complex structure to an equivalent system of single degree of freedom oscillators so this can be applied to the structure as a whole for any selected mode. The response in a given mode i in direction j is

$$x_{ji} = \Gamma_{ij} \frac{a_{spectral}}{(2\pi f_i)^2} \phi_i$$

Where Γ_{ij} is the participation factor to account for the direction of excitation. The term

$$\Gamma_{ji} \frac{a_{spectral}}{(2\pi f_i)^2}$$

is the modal multiplier.

For global x & y we use Γ_x & Γ_y . So for excitation at an angle α we want to use Γ_{α} . Going back to the definition of the participation factor in x and y directions:

$$\Gamma_x = \frac{\phi^T \mathbf{M} \mathbf{x}}{\hat{m}}$$

$$\Gamma_y = \frac{\phi^T \mathbf{M} \mathbf{y}}{\hat{m}}$$

Where x & y corresponds to a rigid body displacement in the x & y directions. So the rigid body vector at α is

$$\mathbf{r}_{\alpha} = x \cos \alpha + y \sin \alpha$$

And the orthogonal direction α' would have a rigid body vector

$$\mathbf{r}_{\alpha'} = -x \sin \alpha + y \cos \alpha$$

This means that for a rotated excitation direction we just need to rotate the participation factors and we don't need to transform the displacements, etc.

$$\Gamma_{\alpha} = \frac{\varphi^T \mathbf{M} \mathbf{x}}{\hat{m}} \cos \alpha + \frac{\varphi^T \mathbf{M} \mathbf{y}}{\hat{m}} \sin \alpha$$

$$\Gamma_{\alpha'} = -\frac{\varphi^T \mathbf{M} \mathbf{x}}{\hat{m}} \sin \alpha + \frac{\varphi^T \mathbf{M} \mathbf{y}}{\hat{m}} \cos \alpha$$

or

$$\Gamma_{\alpha} = \Gamma_x \cos \alpha + \Gamma_y \sin \alpha$$

$$\Gamma_{\alpha'} = -\Gamma_x \sin \alpha + \Gamma_y \cos \alpha$$

That leaves the only transformation we need being the transformation of global displacements to local for nodes in constraint axes. For these we want to transform modal results from global to local, do the combination and transform combined value from local to global.

The modal responses are then combined using one of several combination methods.

Combinations

The main combination methods are:

ABSSUM

$$x = \sum_i |x_i|$$

SRSS

$$x = \sqrt{\sum_i x_i^2}$$

CQC⁵

$$x = \sqrt{\sum_i \sum_j x_i \cdot \rho_{ij} \cdot x_j}$$

where

$$\rho_{ij} = \frac{8\sqrt{\zeta_i \zeta_j} (\zeta_i + \beta_{ij} \zeta_j) \beta_{ij}^{3/2}}{(1 - \beta_{ij}^2)^2 + 4\zeta_i \zeta_j \beta_{ij} (1 + \beta_{ij}^2) + 4(\zeta_i^2 + \zeta_j^2) \beta_{ij}^2}$$

where ζ_i and ζ_j is the damping associated with frequencies f_i and f_j .

$$\beta_{ij} = \frac{f_i}{f_j} \quad f_j \geq f_i$$

⁵ Wilson, der Kiureghian & Bayo, 'Earthquake Engineering and Structural Dynamics', Vol 9, pp 187-194 (1981),

If the damping is constant this simplifies to

$$\rho_{ij} = \frac{8\zeta(1 + \beta_{ij})\beta_{ij}^{3/2}}{(1 - \beta_{ij}^2)^2 + 4\zeta\beta_{ij}(1 + \beta_{ij})^2}$$

Rosenbluth⁶

$$x = \sqrt{\sum_i \sum_j x_i \cdot \rho_{ij} \cdot x_j}$$

where

$$\rho_{ij} = \frac{2\sqrt{\zeta_i \zeta_j} / (\zeta_i + \zeta_j)}{1 + [(f_i - f_j) / (\zeta_i f_i + \zeta_j f_j)]^2}$$

CQC3

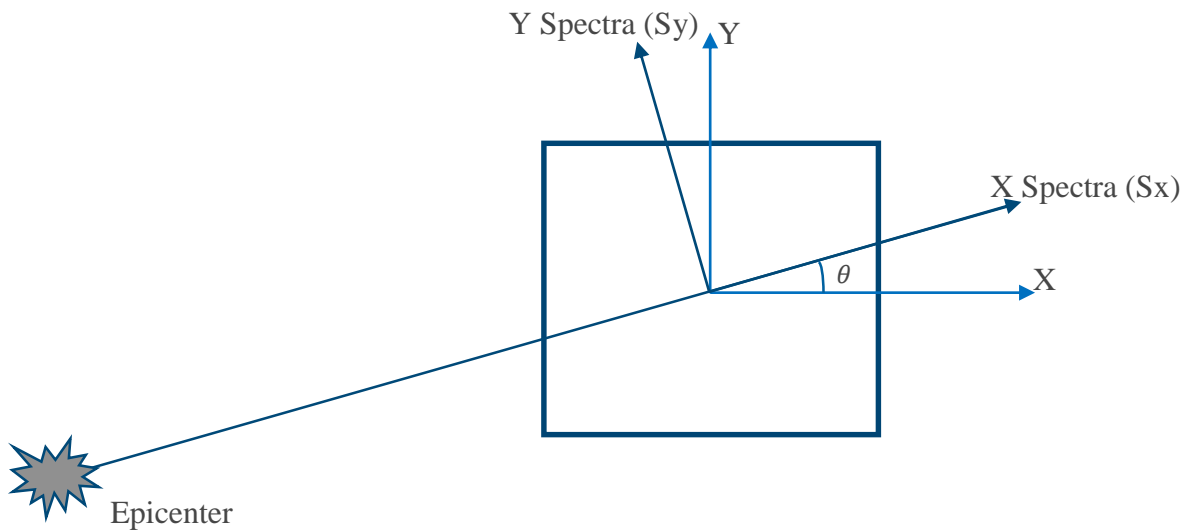
In SRSS method, the spectra s_x, s_y are applied to 100% on the principle directions. The responses obtained from SRSS combination has equal contributions from all the directions. However, in practice the same ground motion will not occurs in both the direction. Therefore, SRSS yields conservative results.

Menun and Der Kiureghian⁷ (1998) presented the CQC3 combination method for combination of the orthogonal spectrum. Let assume s_x, s_y are the major and minor spectra applied at an arbitrary angle θ from the structural axis. To simplify the analysis further assume the s_y spectra is some fraction of s_x spectra.

$$s_y = a \times s_x$$

⁶ ASCE 4-09 Seismic analysis of safety related nuclear structures and commentary', Chapter 4.0 Analysis of Structures (2009)

⁷ Menun, C., and A. Der Kiureghian. 1998. "A Replacement for the 30 % Rule for Multicomponent Excitation," Earthquake Spectra. Vol. 13, Number 1. February.



The peak response value can be estimated using the fundamental CQC3 equation

$$Q = \left[Q_x^2 + a_2 Q_y^2 - (1 - a^2)(Q_x^2 - Q_y^2) \sin^2 \theta + 2(1 - a^2) Q_{xy} \sin \theta \cos \theta + Q_z^2 \right]^{\frac{1}{2}}$$

where

$$Q_x^2 = \sum_i \sum_j q_{xi} \rho_{ij} q_{xj}$$

$$Q_y^2 = \sum_i \sum_j q_{yi} \rho_{ij} q_{yj}$$

$$Q_{xy} = \sum_i \sum_j q_{xi} \rho_{ij} q_{yj}$$

$$Q_z^2 = \sum_i \sum_j q_{zi} \rho_{ij} q_{zj}$$

and q_{xi}, q_{yi} are the modal quantities produced by spectrums applied at x and y directions, q_{zi} is the modal value produced by the vertical spectrum and θ is the arbitrary angle at which the lateral spectra is applied.

Normally, the value of θ is not known. The critical angle that produces maximum response can be calculated using

$$\theta_{cr} = \frac{1}{2} \tan^{-1} \left(\frac{2Q_{xy}}{Q_x^2 + Q_y^2} \right)$$

And the critical response becomes

$$Q = \left[Q_x^2 + a_2 Q_y^2 - (1 - a^2)(Q_x^2 - Q_y^2) \sin^2 \theta_{cr} + 2(1 - a^2) Q_{xy} \sin \theta_{cr} \cos \theta_{cr} + Q_z^2 \right]^{\frac{1}{2}}$$

If the value of $a = 1$, CQC3 combination reduces to SRSS combination. The peak response value is not dependent on the θ and the peak response can be estimated using.

$$Q_{\max} = \sqrt{Q_x^2 + Q_y^2 + Q_z^2}$$

There is no specific guidelines available to choose the value of α . Menun and Der Kiureghian presented an example for CQC3 combination with a value ranging from 0.50 to 0.85.

Storey Inertia Forces

The storey inertia forces can be calculated from the storey mass, m and inertia, I_{zz} , response spectrum and the modal results. The storey modal translations (u_x, u_y) and rotations (θ_z) are calculated (see below)

The force and moment for excitation in the i^{th} direction are then determined from

$$f_x = m \cdot s_{code} \sum_{CQC} (\Gamma_i a_{spec} u_x)$$

$$f_y = m \cdot s_{code} \sum_{CQC} (\Gamma_i a_{spec} u_y)$$

$$M_z = I_{zz} \cdot s_{code} \sum_{CQC} (\Gamma_i a_{spec} \theta_z)$$

Where s_{code} is the code scaling factor, a_{spec} is the spectral acceleration and Γ_i the participation factor.

Equivalent Static and Accidental Torsion Load

Many seismic codes have procedures for calculation of equivalent static or accidental torsion loads.

GSA provides a method of calculating these loads.

The first stage in the calculation is to establish the nodal masses. This includes the mass of elements plus the additional mass derived from any loads. If a modal analysis has previously been carried out this information is picked up from that analysis task. For each storey we can calculate the storey mass M_s by summing the mass of all nodes in that storey.

If a response spectrum case has been selected the base shear, V , is extracted from that calculation, otherwise the base shear is calculated using the code equations (e.g. UBC, IBC or FEMA).

At this point different codes have different requirements. In UBC 1997, depending on the period, an additional force, F_t , is added to the top storey.

$$F_t = 0 \quad T < 0.7s$$

$$F_t = 0.25V \quad 0.07 T > 0.25s$$

$$F_t = 0.07TV$$

For IBC and FEMA an exponent K on the distribution function is required

$$\begin{aligned}
 K &= 1 & T < 0.5s \\
 K &= 2 & T > 2.5s \\
 K &= 1 + \frac{(T - 0.5)}{2}
 \end{aligned}$$

For other codes K is set to 1.

Equivalent Static

We then calculate the force to be applied to each storey, s , at height, h

$$F_s = \frac{M_s H_s^K}{\sum M_s H_s^K} (V - F_t)$$

And for the top storey

$$F_s = \frac{M_s H_s^K}{\sum M_s H_s^K} (V - F_t) + F_t$$

This storey force is then distributed to the nodes, n , in proportion to their mass

$$F_n = \frac{M_n}{M_s} F_s$$

Accidental Torsion

For the accidental torsion we calculate the storey masses as for the equivalent static and we calculate the centre of mass of each storey. Storey calculations are relative to the centre of mass. We also need the width of the storey which is calculated by the difference in the extreme coordinates in the direction of interest.

We have an offset, o , which is based on the width of the storey. The accidental torsion moment for the storey $M_{zz,s}$ is then

$$M_{zz,s} = F_s \cdot o$$

This is then applied as forces to the nodes in the storeys

$$\hat{F}_n = \frac{M_n}{M_s} F_s \cdot o$$

As well as a resulting torsion on the storey this may lead to a force

$$\hat{F} = \sum_{nodes} \hat{F}_n$$

So we correct for this by adjusting these forces by

$$\tilde{F}_n = \hat{F}_n - \frac{M_n}{M_s} \hat{F}_s$$

And sum the moment on the storey is

$$\tilde{M}_{z,z,s} = \sum_{nodes} F_n \cdot o_n$$

Finally we adjust these force values so that we have the correct moment on the storey

$$F_n = \tilde{F}_n \frac{M_{z,z,s}}{\tilde{M}_{z,z,s}}$$

Rotation at the end of a bar (beam)

Force

Point load W at position a

From Roark⁸, the end rotations for a point load are

$$\theta_0 = -\frac{W}{6EI} \frac{a}{l} (2l - a)(l - a)$$

$$\theta_1 = \frac{W}{6EI} \frac{a}{l} (l^2 - a^2)$$

Letting the distance from end 1 be b these equations can be rewritten

$$\theta_0 = -\frac{W}{6EI} \frac{ab(l+b)}{l}$$

$$\theta_1 = \frac{W}{6EI} \frac{ab(l+b)}{l}$$

Varying load from $[a : b]$ with intensity w_a and w_b .

Using the equations above from Roark the end rotations for a point load at x are

$$\theta_0 = -\frac{W}{6EI} \frac{x}{l} (2l - x)(l - x)$$

$$\theta_1 = \frac{W}{6EI} \frac{x}{l} (l^2 - x^2)$$

Using these and integrating over the element gives

$$\theta_0 = -\frac{1}{6EI} \int_a^b w(x) \frac{x}{l} (2l - x)(l - x) dx \quad (1)$$

$$\theta_1 = \frac{1}{6EI} \int_a^b w(x) \frac{x}{l} (l^2 - x^2) dx \quad (2)$$

The load intensity is a linear function in the range $[a : b]$

⁸ Roark Formulas for Stress and Strain, Table 3 (1.e)

$$\begin{aligned}
w(x) &= w_a + (w_b - w_a) \frac{(x-a)}{(b-a)} \\
&= \frac{w_a(b-a)}{(b-a)} + \frac{(w_b - w_a)x}{(b-a)} - \frac{(w_b - w_a)a}{(b-a)} \\
&= \frac{w_a b - w_a a - w_b a + w_a a}{(b-a)} + \frac{(w_b - w_a)x}{(b-a)} \\
&= \frac{w_a b - w_b a}{(b-a)} + \frac{(w_b - w_a)x}{(b-a)}
\end{aligned}$$

Or

$$w(x) = w_p + w_q x \quad (3)$$

where

$$\begin{aligned}
w_p &= \frac{w_a b - w_b a}{(b-a)} \\
w_q &= \frac{(w_b - w_a)}{(b-a)}
\end{aligned}$$

Substituting equation 3 in 1 for end 0

$$\begin{aligned}
\theta_0 &= -\frac{1}{6EI} \int_a^b (w_p + w_q x) \frac{x}{l} (2l-x)(l-x) dx \\
&= -\frac{w_p}{6EI} \int_a^b \frac{x}{l} (2l-x)(l-x) dx - \frac{w_q}{6EI} \int_a^b \frac{x^2}{l} (2l-x)(l-x) dx \\
&= -\frac{w_p}{6EI} \int_a^b x(2l^2 - 3lx + x^2) dx - \frac{w_q}{6EI} \int_a^b x^2(2l^2 - 3lx + x^2) dx \\
&= -\frac{w_p}{6EI} \int_a^b (2l^2 x - 3lx^2 + x^3) dx - \frac{w_q}{6EI} \int_a^b (2l^2 x^2 - 3lx^3 + x^4) dx
\end{aligned}$$

so

$$\begin{aligned}
\theta_0 &= -\frac{w_p}{6EI} \left[l^2 x^2 - lx^3 + \frac{x^4}{4} \right]_a^b - \frac{w_q}{6EI} \left[\frac{2}{3} l^2 x^3 - \frac{3}{4} lx^4 + \frac{x^5}{5} \right]_a^b \\
&= -\frac{w_p}{6EI} \left[l^2 (b^2 - a^2) - l(b^3 - a^3) + \frac{(b^4 - a^4)}{4} \right] \\
&\quad - \frac{w_q}{6EI} \left[\frac{2}{3} l^2 (b^3 - a^3) - \frac{3}{4} l(b^4 - a^4) + \frac{(b^5 - a^5)}{5} \right]
\end{aligned}$$

Substituting equation 3 in 2 for end 1

$$\begin{aligned}
\theta_1 &= -\frac{1}{6EI} \int_a^b (w_p + w_q x) \frac{x}{l} (l^2 - x^2) dx \\
&= -\frac{w_p}{6EI} \int_a^b x (l^2 - x^2) dx - \frac{w_q}{6EI} \int_a^b x^2 (l^2 - x^2) dx \\
&= -\frac{w_p}{6EI} \int_a^b (l^2 x - x^3) dx - \frac{w_q}{6EI} \int_a^b (l^2 x^2 - x^4) dx
\end{aligned}$$

so

$$\begin{aligned}
\theta_1 &= -\frac{w_p}{6EI} \left[\frac{l^2 x^2}{2} - \frac{x^4}{4} \right]_a^b - \frac{w_q}{6EI} \left[\frac{l^2 x^3}{3} - \frac{x^5}{5} \right]_a^b \\
&= -\frac{w_p}{6EI} \left[\frac{l^2 (b^2 - a^2)}{2} - \frac{(b^4 - a^4)}{4} \right] \\
&\quad - \frac{w_q}{6EI} \left[\frac{l^2 (b^3 - a^3)}{3} - \frac{(b^5 - a^5)}{5} \right]
\end{aligned}$$

Moment

Point moment M at position a

From Roark⁹, the end rotations for a point load at a are

$$\theta_0 = -\frac{M}{6EI} (2l^2 - 6al + 3a^2)$$

$$\theta_1 = \frac{M}{6EI} (l^2 - 3a^2)$$

Letting the distance from end 1 be b these equations can be rewritten

$$\theta_0 = \frac{M}{6EI} (l^2 - 3b^2)$$

$$\theta_1 = \frac{M}{6EI} (l^2 - 3a^2)$$

Varying load from $[a : b]$ with intensity m_a and m_b .

Using the equations above from Roark the end rotations for a point moment at x are

$$\theta_0 = -\frac{M}{6EI} (2l^2 - 6xl + 3x^2)$$

⁹ Roark Formulas for Stress and Strain, Table 3 (3.e)

$$\theta_1 = \frac{M}{6EI}(l^2 - 3x^2)$$

Using these and integrating over the element gives

$$\theta_0 = -\frac{1}{6EI} \int_a^b m(x)(2l^2 - 6xl + 3x^2) dx \quad (4)$$

$$\theta_1 = \frac{1}{6EI} \int_a^b m(x)(l^2 - 3x^2) dx \quad (5)$$

As with the forces the moment intensity can be written as

$$m(x) = m_p + m_q x \quad (6)$$

where

$$m_p = \frac{m_a b - m_b a}{(b - a)}$$

$$m_q = \frac{(m_b - m_a)}{(b - a)}$$

Substituting equation 6 in 4 for end 0

$$\begin{aligned} \theta_0 &= -\frac{1}{6EI} \int_a^b (m_p + m_q x)(2l^2 - 6xl + 3x^2) dx \\ &= -\frac{m_p}{6EI} \int_a^b (2l^2 - 6xl + 3x^2) dx - \frac{m_q}{6EI} \int_a^b (2l^2 x - 6x^2 l + 3x^3) dx \end{aligned}$$

so

$$\begin{aligned} \theta_0 &= -\frac{m_p}{6EI} \left[2l^2 x - 3x^2 l + x^3 \right]_a^b - \frac{m_q}{6EI} \left[l^2 x^2 - 2x^3 l + \frac{3x^4}{4} \right]_a^b \\ &= -\frac{m_p}{6EI} \left[2l^2(b-a) - 3(b^2 - a^2)l + (b^3 - a^3) \right] \\ &\quad - \frac{m_q}{6EI} \left[l^2(b^2 - a^2) - 2(b^3 - a^3)l + \frac{3(b^4 - a^4)}{4} \right] \end{aligned}$$

Substituting equation 6 in 5 for end 1

$$\begin{aligned} \theta_1 &= \frac{1}{6EI} \int_a^b (m_p + m_q x)(l^2 - 3x^2) dx \\ &= \frac{m_p}{6EI} \int_a^b (l^2 - 3x^2) dx + \frac{m_q}{6EI} \int_a^b (l^2 x - 3x^3) dx \end{aligned}$$

so

$$\begin{aligned}\theta_1 &= \frac{m_p}{6EI} \left[l^2 x - x^3 \right]_a^b + \frac{m_q}{6EI} \left[\frac{l^2 x^2}{2} - \frac{3x^4}{4} \right]_a^b \\ &= \frac{m_p}{6EI} \left[l^2 (b-a) - (b^3 - a^3) \right] \\ &\quad + \frac{m_q}{6EI} \left[\frac{l^2 (b^2 - a^2)}{2} - \frac{3(b^4 - a^4)}{4} \right]\end{aligned}$$

Thermal

For a thermal gradient ϕ applied to a beam the curvature is

$$\kappa = \alpha\phi$$

So the radius of curvature is

$$r = \frac{1}{\kappa}$$

Assuming a circular arc the rotation at the end is perpendicular to the radial line of the arc. This means that the angle of the radial line to the beam original configuration is

$$\begin{aligned}\cos \beta &= \frac{l/2}{r} \\ \beta &= \cos^{-1} \left(\frac{l\kappa}{2} \right)\end{aligned}$$

And

$$\begin{aligned}\varphi &= \frac{\pi}{2} - \cos^{-1} \left(\frac{l\kappa}{2} \right) \\ \varphi &= \frac{\pi}{2} - \cos^{-1} \left(\frac{l\alpha\phi}{2} \right)\end{aligned}$$

Pre-stress

Only the tendon per-stress P with an offset o will give rise to a rotation. The effect of this is a uniform moment over the length of the element of

$$m = Po$$

This case can then be treated as above for a patch moment.

Beam distortion

A beam distortion can be a displacement discontinuity or a rotation discontinuity. For a displacement discontinuity of v at x the rotation is then

$$\theta_0 = \theta_1 = \frac{v}{l}$$

For a rotational discontinuity of φ at l the rotation is calculated by defining a displacement of h at l then the angle α and end 0 and β at end 1 gives the set of equations

$$b = l - a$$

$$\varphi = \alpha + \beta$$

$$\tan \alpha = \frac{\sin \alpha}{\cos \alpha} = \frac{h}{a}$$

$$\tan \beta = \frac{\sin \beta}{\cos \beta} = \frac{h}{b}$$

Substituting for β in the last equation gives

$$b \sin(\varphi - \alpha) = h \cos(\varphi - \alpha)$$

$$b(\sin \varphi \cos \alpha - \cos \varphi \sin \alpha) = h(\cos \varphi \cos \alpha + \sin \varphi \sin \alpha)$$

$$b \left(\sin \varphi - \cos \varphi \frac{\sin \alpha}{\cos \alpha} \right) = h \left(\cos \varphi + \sin \varphi \frac{\sin \alpha}{\cos \alpha} \right)$$

Substituting for terms in α

$$b(a \sin \varphi - h \cos \varphi) = h(a \cos \varphi + h \sin \varphi)$$

$$\sin \varphi h^2 + (a + b) \cos \varphi h - ab \sin \varphi$$

Then the rotation angles at end 0 and 1 are

$$\alpha = \tan^{-1} \left(\frac{h}{a} \right)$$

$$\beta = \tan^{-1} \left(\frac{h}{b} \right)$$

Second Moments of Area & Bending

The second moments of area are defined as

$$I_{yy} = \int_A z^2 dA$$

$$I_{zz} = \int_A y^2 dA$$

$$I_{yz} = \int_A yz dA$$

For symmetric sections I_{xy} is zero.

For uniaxial bending or bending about principal axes

$$M_y = EI_{yy} \kappa_y$$

$$M_z = EI_{zz} \kappa_z$$

When there is biaxial bending these have to be modified to

$$\begin{Bmatrix} M_y \\ M_z \end{Bmatrix} = E \begin{bmatrix} I_{yy} & -I_{yz} \\ -I_{yz} & I_{zz} \end{bmatrix} \begin{Bmatrix} \kappa_y \\ \kappa_z \end{Bmatrix}$$

As the second moments of area form a tensor these can be rotated to different axes using a rotation matrix

$$\begin{bmatrix} \tilde{I}_{yy} & -\tilde{I}_{yz} \\ -\tilde{I}_{yz} & \tilde{I}_{zz} \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} I_{yy} & -I_{yz} \\ -I_{yz} & I_{zz} \end{bmatrix} \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix}$$

Or

$$\tilde{I}_{yy} = I_{yy} \cos^2 \theta + I_{zz} \sin^2 \theta - 2I_{yz} \sin \theta \cos \theta$$

$$\tilde{I}_{zz} = I_{yy} \sin^2 \theta + I_{zz} \cos^2 \theta + 2I_{yz} \sin \theta \cos \theta$$

$$\tilde{I}_{yz} = I_{yy} \sin \theta \cos \theta - I_{zz} \sin \theta \cos \theta + I_{yz} (\cos^2 \theta - \sin^2 \theta)$$

Or in terms of double angles

$$\tilde{I}_{yy} = \frac{I_{yy} + I_{zz}}{2} + \frac{I_{yy} - I_{zz}}{2} \cos 2\theta - I_{yz} \sin 2\theta$$

$$\tilde{I}_{zz} = \frac{I_{yy} + I_{zz}}{2} - \frac{I_{yy} - I_{zz}}{2} \cos 2\theta + I_{yz} \sin 2\theta$$

$$\tilde{I}_{yz} = \frac{I_{yy} - I_{zz}}{2} \sin 2\theta + I_{yz} \cos 2\theta$$

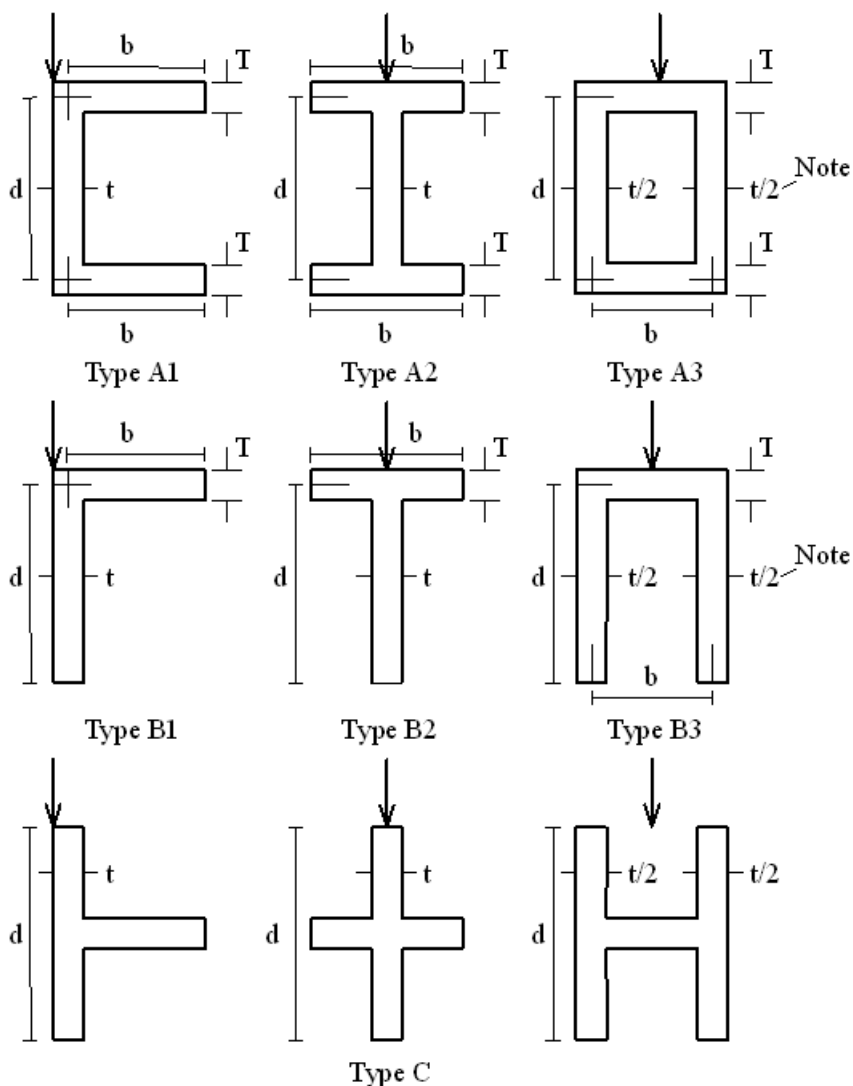
Shear Areas

Thin-walled Sections

The effect of shear deformation on the results of a structural analysis is usually negligible. Where it is more significant, it will usually suffice to make a simple approximation to the shear deformation area of members with a cross-section such as those shown in Fig 1. The usual approximation is, by analogy with a simple rectangular beam, to take $\frac{5}{6}$ or $\frac{1}{2}$ of the total web area (overall depth \times web thickness).

For those rare structures where the shear deformation is very important it may be necessary to use a more exact value for the area. This Note gives formulae for F for the cross-sections of Fig.1 where

$$A_s = F \frac{dt}{6}$$



These were derived from the virtual work formula, shear deflection per unit length = $\int \frac{q^2}{Gt} ds$.

Here q is the shear flow at any point at the middle of the wall thickness, the shear stress q/t is assumed constant across the wall thickness t , and the integration extends over the whole cross-section.

To see what the formulae mean in practice, they were applied to steel sections taken from the handbook with the results shown in the table below. In a web or flange with varying thickness, t was assumed constant at its average value. The smaller values of F in a range correspond to cross-sections with squarer aspect ratios. Samples only of UBs, UCs and channels were taken.

It can be seen that, for sections with top and bottom flanges bending, as nature intended, in their strong direction, the usual approximation is satisfactory (although it should be noted that d is the distance between flange centres, not overall depth). For the more bizarre sections used in bending, values of F are seen to be lower than expected, especially when they are perversely bent in their weakest direction.

Section	F	
Bending in strong direction		
UB	5.72 to 5.81	
Joist	5.17 to 5.78	
UBP	5.25 to 5.28	
UC	5.28 to 5.53	
SHS	5.0	
RHS	5.0 to 5.49	$d/b = [1 : 2]$
Channel	5.06 to 5.60	
Angle	3.68 to 4.62	$d/b = [1 : 2]$
Tee from UB	4.78 to 4.97	
Tee from UC	4.11 to 4.35	
Cruciform	5	
Bending in weak direction		
Channel	2.12 to 3.78	
RHS	4.02 to 5.0	$d/b = [1 : 2]$
Angle	2.55 to 3.68	(for $d/b = 0.5$ to 1)
I's, T's, & cruciform	5.0	

With regard to calculating shear stresses, the exact distribution is not normally required, or even usable, because Codes of Practice base the shear strength on an allowable average shear stress calculated on the total net area Dt . However, the shear distribution is sometimes required to design welds or concrete stitches, and, since it was found in the process of deriving the formulae for F , formulae for the stress factors k_i and k_m are given. Here,

$$q_i = k_i \frac{V}{d}$$

is the shear flow in the web at the junction with the flange, and

$$q_{mi} = k_m \frac{V}{d}$$

is the maximum shear flow in the web, in which V is the shear force at the section.

For circular annuli, assuming that the stress is constant across the wall thickness t , both the deflection and maximum stress can be obtained using a shear area of half the actual area, that is $\pi r t$ where r is the mean radius.

Formulae

$$\lambda \equiv \frac{dt}{bT} \quad \beta \equiv \frac{bt}{dT} \quad \phi \equiv \frac{d}{b}$$

Shear deformation area

$$A_s = F \frac{dt}{6}$$

Type A1

$$F = \frac{(6 + \lambda)^2}{4\beta + 6 + 2\lambda + 0.2\lambda^2}$$

Types A2 & A3

$$F = \frac{(6 + \lambda)^2}{\beta + 6 + 2\lambda + 0.2\lambda^2}$$

Special case of A3 with constant wall thickness so that $T = t$, $\lambda = 2\phi$, $\beta = 2/\phi$

$$F = \frac{10\phi(3 + \phi)^2}{5 + 15\phi + 10\phi^2 + 2\phi^3}$$

Type B1

$$F = \frac{(4 + \lambda)^2}{2\beta + 3.2 + 1.4\lambda + 0.2\lambda^2}$$

Special case of B1 with constant $T = t, \lambda = \phi, \beta = 1/\phi$

$$F = \frac{5\phi(4 + \phi)^2}{(1 + \phi)(10 + 6\phi + \phi^2)}$$

Types B2 & B3

$$F = \frac{(4 + \lambda)^2}{0.5\beta + 3.2 + 1.4\lambda + 0.2\lambda^2}$$

Special case of B3 with constant thickness $T = t/2, \lambda = 2\phi, \beta = 2/\phi$

$$F = \frac{20\phi(2 + \phi)^2}{5 + 16\phi + 14\phi^2 + 4\phi^3}$$

Type C

$$F = 5$$

Stress Factors

Type A1, A2 & A3

$$k_j = \frac{6}{6 + \lambda} \quad k_m = \frac{3}{2} \left[\frac{4 + \lambda}{6 + \lambda} \right]$$

Type B1, B2 & B3

$$k_j = \frac{6}{4 + \lambda} \quad k_m = \frac{3}{2} \left[\frac{(2 + \lambda)^2}{(1 + \lambda)(4 + \lambda)} \right]$$

Solid Sections

For rectangular beams it is usually sufficiently accurate to take the shear area for deflection as $\frac{5}{6}bd$ where b is the breadth and d the depth of the section. The corresponding maximum shear stress is $\frac{3}{2}V/bd$. It should be noted however that for wide beams the maximum shear stress is underestimated by this formula: for a beam with an aspect ratio of 1 the maximum stress is 12.6% higher. (For a beam with an aspect ratio of 50 (for example a slab) the maximum stress is about 2000% higher but, as this is a Poisson's ratio effect, it is difficult to believe that this has any practical significance!¹⁰)

¹⁰ Timoshenko, S.P. and Goodier, J.N. Theory of elasticity. 3rd edition. McGraw-Hill, 1970.

For circular sections the shear area for deflections is

$$\frac{6\pi r^2}{7 + \frac{\nu^2}{(1+\nu)^2}}$$

where ν is Poisson's ratio and r the radius. The expression is very insensitive to the value of ν . The maximum shear stress is given by

$$\left[\frac{1.5 + \nu}{1 + \nu} \right] \frac{V}{\pi r^2}$$

which varies from

$$1.5 \frac{V}{\pi r^2} \quad \text{for } \nu = 0$$

to

$$1.33 \frac{V}{\pi r^2} \quad \text{for } \nu = 0.5$$

with

$$1.38 \frac{V}{\pi r^2} \quad \text{for } \nu = 0.3$$

This item was written by John Blanchard and Ian Feltham for Feedback Notes [an Ove Arup & Partners internal publication] (1992 NST/21) originally published in October 1992. Incorporates 1996NST/10 and is reproduced here with permission.

Storey Displacements

Storey displacement are calculated from

$$u_s = \frac{1}{n} \sum_i u_i$$

Where i are the nodes in the storey and n is the number of nodes in the storey. The rotations are the calculated relative to the centre of mass, c_m .

The position of node i relative to the centre of mass is

$$c = c_i - c_m$$

the distance from the centre of mass is

$$r = \sqrt{c_x^2 + c_y^2}$$

and the component of displacement giving rise to rotation is

$$u = u_i - u_s$$

The rotation of the storey is then defined as

$$\theta = \frac{1}{n} \sum_i \left(\frac{u_y c_x - u_x c_y}{r^2} \right)$$

Sub-model Extraction

This feature allows a sub-model to be extracted from GSA, so that the sub-model can be investigated in more detail. There are two aspects to the sub-model: the elements that form the sub-model and the tasks that are to be associated with the sub-model.

The nodes that form the sub-model and the nodes that form the boundary between the sub-model and the remainder of the model are identified. A new model is then created for the sub-model. Nodes on the boundary are fully restrained. Properties are copied directly. Constraints and loading are updated to include only those associated with the sub-model. Tasks are updated based on the tasks selected for the sub-model.

Static analysis tasks

For static analysis tasks the displacements at the boundary nodes are extracted for each analysis case. These are then applied in a new load case as settlements, and the analysis cases updated to include these settlements.

Modal dynamic and Ritz analysis cases

For dynamic tasks there is the option of Local or Global response.

Local

If the mode is predominantly local it may be more appropriate to consider the boundary of the sub model as fixed and to carry out a modal analysis of the sub-model. In this case the model extraction is straightforward. In this case the analysis task is copied directly

Global

It is not possible to carry out a sub-model model analysis when the mode is global. In this case the modal results are used to create a set of static loads

The modal analysis of the full model gives us eigenvalues and eigenvectors which satisfy

$$K\varphi_i - \lambda_i M\varphi_i = 0$$

This can be rearranged in the form

$$K\varphi_i = \lambda_i M\varphi_i$$

Which we can consider as a static (pseudo modal) analysis of

$$K\varphi_i = p_i$$

where

$$p_i = \lambda_i M\varphi_i$$

When extracting the sub-model the frequency (eigenvalue) and mode shape (eigenvector) can be used to create a set of node loads:

$${}^n p_i = \lambda_i {}^n M {}^n \varphi_i$$

$${}^n p_i = (2\pi f_i)^2 \cdot {}^n M {}^n \varphi_i$$

The modal task in the full model is mapped to a static task in the sub-model. As for the static analysis the displacements at the boundary nodes are used to determine settlements at the boundary nodes and the inertia loads are saved as node loads. The static analysis of these loads will then recover the mode shapes of the original model.

Response spectrum analysis

Response spectrum analysis is a combination of modal results scaled to match a given response spectrum. Extracting the modal results as static load cases means that the dynamic details are lost to the sub-model. To overcome this problem the response spectrum tasks are mapped to pseudo response spectrum tasks. A pseudo response spectrum task assumes that a static analysis case represents a mode shape. The dynamic details are supplied directly to the task in the form of frequency, modal mass and effective masses. When a response spectrum task is extracted from the full model these details are recovered from the modal analysis and included in the pseudo response spectrum analysis task.

Given a shape from a pseudo modal analysis the scaling for each mode i , for excitation in the j direction is:

$$u_i = \Gamma_{i,j} \frac{a_{spectral}}{(2\pi f_i)_2} \varphi_i$$

Where

$$\Gamma_{i,j} = \sqrt{\frac{\mu_{i,j}}{\tilde{m}}}$$

If the pseudo modal analysis is carried out using the static procedure above, and the frequency and participation factor are known from the master model, then the modal contribution to the response spectrum analysis can be estimated. Thereafter the combination is just as for any normal modal analysis.

Torce Lines

GSA gives the option of plotting 'Torce lines', which are similar to thrust lines but include the effects of torsion. They are equal for plane frames.

Good structural design requires a clear thinking head, well-presented information and some creative flair. At the detailed level designs progress by iterative evolution and the quality of the design depends on the effectiveness of the interaction between the designer and the information which describes the behaviour of the structure in the current cycle. In today's world this information is almost always presented by a computer. Graphical plots of the parameters used in the numerical analyses are often ergonomically inefficient, meaning that although they contain the information they fail to transmit insight or understanding into the head of the designer. Better representations are needed.

For many applications thrust lines meet this need. They are useful where members are subjected to combined axial load and bending, particularly for compression members made of a material that takes no tension. In these cases the adequacy of the member can often be reasonably described by a limit on the eccentricity of the thrust line. Thrust lines are frequently used by the designers of masonry arches.

A thrust line is the locus along a member of points drawn at an eccentricity of

$$\frac{M}{F_x}$$

from the centroidal axis of the member, where M is the moment and F_x the axial force. It follows the line of action of the force carried by the member and two related points follow from this:

- No scale is needed when drawing a thrust line. It is not a diagram superimposed on a view of a structure. It occupies the position where it is shown.
- It is not subject to any sign conventions.

However the key feature of the thrust line which makes it so useful is that it is a complete description of the forces carried by the member. Because of this completeness property a designer can, with no loss of accuracy, substitute a thrust line, which he or she can visualise, for the combination of the three numbers representing the axial force, moment and shear, which remains stubbornly abstract.

In a computer model of a structure the thrust line is derived from the forces in a member but because it occupies the actual position of the force transmitted through the structure being modelled it follows that its position remains fixed in space even if the member which 'supports' it is moved, providing the movement of the member does not change the force being transmitted. This can be referred to as the invariance property of thrust lines.

The above statements refer to thrust lines in two dimensions, as in a plane frame model. Thrust lines can also be drawn in three dimensions but unfortunately in shifting up a dimension they lose both their completeness and invariance properties and so they lose their usefulness to the designer.

The general state of force in a section of a member in three dimensions cannot be reduced to a single force, but it can be reduced to a combination of a force and a torque or, more completely, to any one of an infinite number of force/torque pairs. There is just one force/torque pair, hereafter referred to as the torque, for which the force axis and the torque axis are parallel. In three dimensions it is torque lines that have the completeness and invariance properties and so it is torque line which can be used by designers to visualise structural actions.

The position of a torque line is derived from the forces in a section of a member as follows

The resultant shear force, Q , and its angle to the z axis, α , are given by

$$Q = \sqrt{F_y^2 + F_z^2}$$

$$\alpha = \tan^{-1}\left(\frac{F_z}{F_y}\right)$$

The angle β between the member axis and the thrust line is given by

$$\beta = \tan^{-1}\left(\frac{Q}{F_x}\right)$$

The torque, T , of the torque is resolved into moments about the three member axes. The component about the x axis is the torque M_x in the member. The other two components modify the bending components M_y and M_z .

Because the force components of the torque (as opposed to the moment components) are the same as those on the member, the thrust line and the torque are parallel.

Hence

$$T \cos \beta = M_x$$

Therefore

$$T = M_x \sec \beta$$

Eccentricities of the torque, t_y and t_z , are given by

$$t_y = -\frac{M_z - T \sin \beta \sin \alpha}{F_x}$$

$$t_z = \frac{M_y - T \sin \beta \sin \alpha}{F_x}$$

hence

$$t_y = -\frac{M_z - M_x \cdot F_z / F_x}{F_x}$$

$$t_z = \frac{M_y - M_x \cdot F_y / F_x}{F_x}$$

How do torques behave? A torque is a unique representation of the force state in a section of structure. Torques can be added or subtracted. There is always a unique result. Note that if two torques lie in the same plane their addition or subtraction is not necessarily in that plane. In three dimensions four forces on a body in equilibrium are necessarily coincident. The same is not true for torques. Consider four coincident torques in equilibrium. If one of the torques is translated without changing its direction then the body is subjected to a moment. Equilibrium can be restored by adjusting the torques in the other three torques. Hence equilibrium is achieved with four non-coincident torques.

Often it will be known that one or more of the torques have zero torque. For example gravity imposes zero torque on a body. How many torqueless torques are needed to restore the coincidence rule? The answer is four. This is the same as the four forces on a body rule. If just one of the four carries a torque it can be resisted by a combination of lateral shifts of the other three forces and coincidence is lost. The usefulness of the concept of coincidence is limited to statements such as: The sum of two coincident torques, one of which is torqueless, is coincident with its components.

For the designer the lesson from this is that, unless a structure is conceived as being truly three dimensional, it is often better to analyse it as a two dimensional plane frame during the design evolution phase so the coincidence rule can be used to understand what is driving the magnitudes of the forces.

This item was written by Angus Low for Feedback Notes [an Arup internal publication] (1999 NST/7) and is reproduced here with permission

Torsion Constant

In structural elements capable of sustaining torsional moments it is necessary to define a constant to specify the 'twisting stiffness'.

This term is referred to as the Torsion Constant. The following notes offer guidance on how to calculate the Torsion Constant. More detailed information may be found in the standard texts.

In the literature there is some confusion over the symbol used to represent the Torsion Constant. In many texts it is referred to as J . Elsewhere the symbol C is used in an effort to distinguish the Torsion Constant from the polar moment of inertia. In GSA and in these notes the symbol J is adopted.

If a circular bar of constant cross section and of length l is subjected to a constant torque T , the angle of twist θ between the ends is

$$\theta = \frac{Tl}{GI}$$

where G is the shear modulus and I the polar moment of inertia.

When the cross section of the bar is non-circular, plane cross sections do not remain plane after deformation and warping will occur. Nevertheless the above equation can still be used with good accuracy, but I should be taken as the appropriate Torsion Constant J as defined below.

Saint Venant's Approximation

Saint Venant represented the torsion constant (J) of a solid section by a function relating the already known characteristic values of a cross section thus

$$\begin{aligned} J &= f(A, I) \\ &= \frac{1}{K} \frac{A^4}{I} \end{aligned}$$

For narrow rectangles ($b_{\max} \times b$) (i.e. thin plates):

$$J = \frac{1}{36} \frac{A^4}{I}$$

which reduces to

$$J = \frac{1}{3} b^3 b_{\max}$$

For circular sections

$$J = \frac{\pi r^4}{2}$$

which is the polar moment of inertia.

It can be shown that for members composed of thin rectangles, the torsion constant is equal to the sum of the J -values of component rectangles, except when the section is closed or 'hollow'.

Rectangular Sections

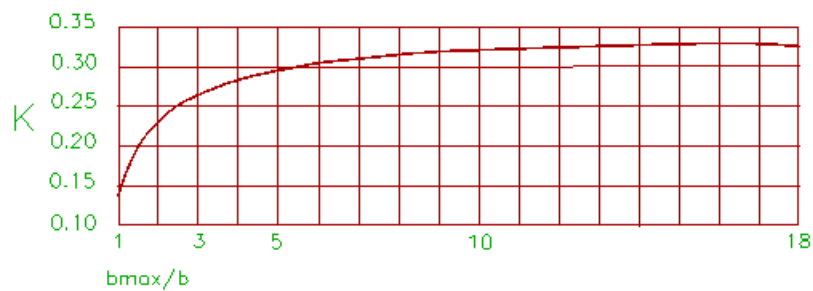
The torsion constant J is given by

$$J = K b^3 b_{\max}$$

where K is a constant depending on the ratio of b_{\max}/b , which can be read either from the table below. Linear interpolation may be used for intermediate values.

or from the graph

b_{\max}/b	K	b_{\max}/b	K	b_{\max}/b	K	b_{\max}/b	K
1.0	0.141	1.5	0.196	2.8	0.258	10.0	0.312
1.1	0.154	1.8	0.216	3.0	0.263	∞	0.333
1.2	0.166	2.0	0.229	4.0	0.281		
1.3	0.175	2.3	0.242	5.0	0.291		
1.4	0.186	2.5	0.249	7.5	0.305		



Note that K converges to $1/3$ for narrow rectangles.

Alternatively

$$K = \frac{1}{3} \left[1 - 0.63 \frac{b}{b_{\max}} \left(1 - \frac{b^4}{12b_{\max}^4} \right) \right]$$

generally, or if $b_{\max} \geq 2b$ then

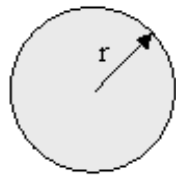
$$\left(1 - \frac{b^4}{12b_{\max}^4} \right) \approx 1$$

and so

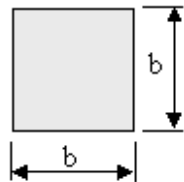
$$K = \frac{1}{3} \left[1 - 0.63 \frac{b}{b_{\max}} \right]$$

Other Sections

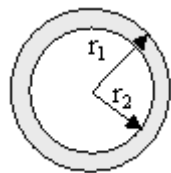
The following table from Ghali and Neville (1978) gives values of the torsion constant (J) for various cross-sectional shapes.



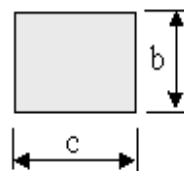
$$J = \frac{\pi r^4}{2}$$



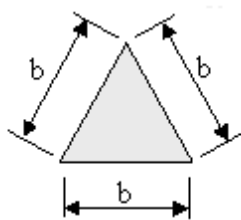
$$J = 0.1406b^4$$



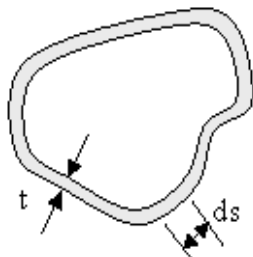
$$J = \frac{\pi(r_1^4 - r_2^4)}{2}$$



$$J = \left[\frac{1}{3} - 0.21 \frac{b}{c} \left(1 - \frac{b^4}{12c^4} \right) \right]$$

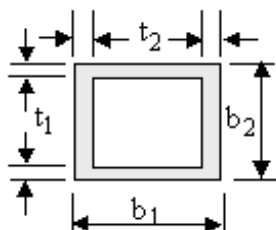


$$J = \frac{\sqrt{3}b^4}{80}$$

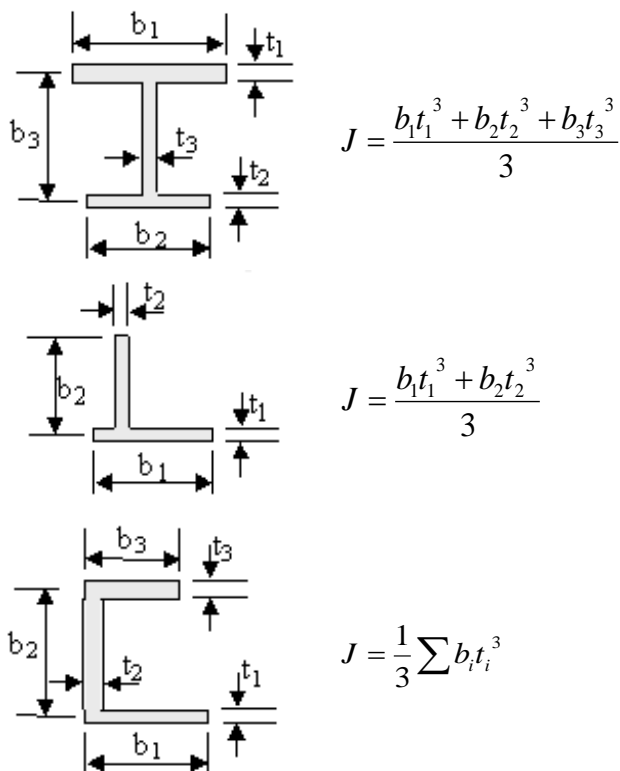


$$J = \frac{4a^2}{\oint \frac{1}{t} ds}$$

where a is the area enclosed by a line through the centre of the thickness and the integral is carried out over the circumference



$$J = \frac{2t_1 t_2 (b_1 - t_2)^2 (b_2 - t_1)^2}{b_1 t_2 + b_2 t_1 - t_2^2 - t_1^2}$$



All the above assumes that the material is linear elastic.

Factored Values for Concrete

In the case of concrete the Torsion Constant (J) needs to be modified for the following reasons:

- The above formulae do not consider cracking of the concrete.
- There is the practical difficulty of reinforcing for torsion.

For concrete, the actual value ranges from 0.15 to 0.5 of the theoretical (linear elastic) value. BS8110 Part 2 Clause 2.4.3 recommends that half the linear elastic value (calculated above) is used in an analysis.

Transformations

There are two type of transformation in GSA: transformation of coordinates from global to local axes and vice versa, and the transformation of results such as displacements, forces and stresses. The coordinate transformation requires a rotation + translation while the displacement transformation requires only a rotation.

The rotation can be defined by a 3×3 direction cosine matrix, so the transformation is

$$\begin{Bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \end{Bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \end{Bmatrix}$$

Or

$$\tilde{\mathbf{x}} = \mathbf{C} \mathbf{x}$$

And the reverse transformation is then

$$\mathbf{x} = \mathbf{C}^{-1} \tilde{\mathbf{x}} = \mathbf{C}^T \tilde{\mathbf{x}}$$

The axis transformation can be considered as

$$\tilde{\mathbf{x}} = \mathbf{C}(\mathbf{x} - \mathbf{o})$$

And the reverse transformation as

$$\mathbf{x} = \mathbf{C}^T \tilde{\mathbf{x}} + \mathbf{o}$$