SUBOUTINE ALGOR - MANUAL PAGE

DESCRIPTION
Sets equation resolution index (flag) according to chosen algorithm

This routine sets the equation resolution index KRESL according to the selected algorithm for solution of the non-linear equilibrium problem.

ARGUMENT LIST

Type	Name		Description
integer	IINCS	>	Current load increment number.
integer	IITER	>	Current equilibrium iteration number.
integer	KRESL	<	Equation resolution index.
integer	KLUND	>	Unloading flag.

SUBOUTINE ARCLEN - MANUAL PAGE

DESCRIPTION
Computes the iterative displacements for the Arc-Length method

This routine updates the global array of iterative nodal displacements using the cylindrical Arc-Length method. Prior to the computation of the iterative displacement, the incremental and total load factors are computed according to the cylindrical arc-length equations. If the evaluation of the load factors fail, i.e., if there are no real roots to the arc-length constraint equation, the return value of the logical argument INCCUT is set to .TRUE. which will activate increment cutting in the main program.

ARGUMENT LIST

Type	Name		Description
double precision	DFACT	<>	Incremental load factor.
double precision	DLAMD	<	Iterative load factor.
double precision	DLENG	<>	Arc-length.
double precision	DLENM	<	Maximum permissible Arc-length. Set only in the first iteration of the first increment.
double precision	DLENP	>	Maximum Arc-length parameter. Needed only in the first iteration of the first increment.
integer	IFNEG	>	Signum (-1/+1) of the determinant of the global tangent stiffness matrix. Required in the first iteration of every load increment to define the sign of the incremental load factor.
integer	IINCS	>	Current load increment number.
integer	IITER	>	Current equilibrium iteration number.
logical	INCCUT	<	Increment cutting flag. Set to .TRUE. on return if no real roots exist for the arc-length constraint equation. This will activate the increment cutting procedure in the main program. Set to .FALSE. otherwise.
double precision	TFACT	<>	Total load factor.

SUBOUTINE ARRGO2 - MANUAL PAGE

DESCRIPTION
Arrange a fourth order tensor in matrix form with G matrix ordering

This routine re-arranges a given fourth order tensor, stored as a 4-index array, in matrix form (2-index array) using G matrix component ordering.
Implemented only for 2-D tensors.

ARGUMENT LIST

Type	Name		Description
double precision	A4TH	>	Fourth order tensor stored as a 4-index array.
double precision	AMATX	<	2-index array containing the components of the given 4th order tensor stored using G matrix ordering.

SUBOUTINE ATMDFB - MANUAL PAGE

DESCRIPTION

Computes the additional tangent modulus, q, for F-Bar elements

This routine computes the additional tangent modulus, q, required to assemble the tangent stiffness matrix for F-bar elements. This routine contains the plane strain and axisymmetric implementations.

ARGUMENT LIST

Type	Name		Description
double precision	AMATX	>	Matrix of components of the current spatial tangent modulus, a.
integer	NTYPE	>	Stress state type flag.
double precision	QMATX	<	The additional tangent modulus required to assemble the tangent stiffness matrix for F-bar elements.
double precision	STRES	>	Array of current Cauchy stress tensor components.

SUBOUTINE BETRIA - MANUAL PAGE

DESCRIPTION
Computes the left Cauchy-Green strain tensor

Given the previous elastic logarithmic strains and the incremental deformation gradient between the previous and current configuration, this routine computes the current elastic trial left Cauchy-Green strain tensor.
This routine contains the plane strain, plane stress and axisymmetric implementations.

ARGUMENT LIST

Type	Name		Description
double precision	BETRL	<	Array of components of the current elastic trial (or total) left Cauchy-Green strain tensor.
double precision	EEN	>	Array of previous elastic engineering logarithmic strains.
double precision	FINCR	>	Incremental deformation gradient between the previous and current configuration.
integer	NTYPE	>	Stress state type. Present routine is compatible with NTYPE=1 (plane stress), NTYPE=2 (plane strain) and NTYPE=3 (axisymmetric condition).

SUBOUTINE CHECK2 - MANUAL PAGE

DESCRIPTION
Performs further checks on the input data.

ARGUMENT LIST

Type	Name	Description
integer	MXFRON <	Maximum frontwidth encountered in the solution of the linear finite element equation system.

SUBOUTINE CHKNDB - MANUAL PAGE

DESCRIPTION
Checks whether a set of nodes matches one of the element boundaries

This routine checks whether a given set of local node numbers match one of the boundaries of an element. If the given set matches one boundary, NODCHK returns with the appropriate node number ordering for numerical integration on that boundary.

ARGUMENT LIST

Type	Name		Description
logical	FOUND	<	Set to .TRUE. if the given set of local node numbers matches one of the boundaries of the element. Set to .FALSE. otherwise.
integer	NNODE	>	Total number of nodes of the element.
integer	NEDGEL	>	Total number of edges (facets in 3-D) of the element.
integer	NODCHK	<>	On entry, defines the set of local node numbers to be matched by a boundary: If local node INODE is part of the set, then NODCHK(INODE)=1. NODCHK(INODE)=0 otherwise. This array returns with the sequence of local node numbers on the matching boundary.
integer	NORDEB	>	Array containing the sequence of local node numbers in each edge (facets in 3-D) of the element.

SUBOUTINE CONVER - MANUAL PAGE

DESCRIPTION
Checks for convergence of the equilibrium iterations

This routine computes the global residual (out-of-balance) force vector and its relative norm and checks whether the convergence criterion has been satisfied.
If convergence has been attained the flag logical CONVRG returns as .TRUE., otherwise it returns as .FALSE.
Also, if divergence is detected, the flag DIVERG returns as .TRUE. (and .FALSE. otherwise).

ARGUMENT LIST

Type	Name		Description
logical	CONVRG	<	Convergence flag.
logical	DIVERG	<	Divergence flag.
integer	IITER	>	Current equilibrium iteration number.
double precision	TOLER	>	Equilibrium convergence tolerance.
double precision	TFACT	>	Current total load factor.

SUBOUTINE CSTEP2 - MANUAL PAGE

DESCRIPTION
Consistent spatial tangent modulus for finite elasto-plastic models

This routine computes the spatial tangent modulus, a, for hyperelastic based (logarithmic strain-based) finite elasto-plasticity models in 2-D: Plane strain, plane stress and axisymmetric states.

ARGUMENT LIST

Type	Name		Description
double precision	AMATX	<	Matrix of components of the spatial tangent modulus, a.
double precision	BETRL	>	Array of components of the elastic trial left Cauchy-Green strain tensor.
double precision	DMATX	>	Array of components of the infinitesimal consistent tangent modulus.
double precision	STRES	>	Array of Cauchy stress tensor components
double precision	DETF	>	Determinant of the current deformation gradient.
integer	NTYPE	>	Stress state type flag.

SUBOUTINE CSTOGD - MANUAL PAGE

DESCRIPTION

Spatial elasticity tensor for the Ogden hyperelastic material model

This routine computes the spatial elasticity tensor, a, for the regularised Ogden hyperelastic material model under plane stress, plane strain and axisymmetric conditions. Note that under plane stress the model implemented here is exactly incompressible, whereas under plane strain and axisymmetric conditions the regularised (penalty type) approach is used to enforce incompressibility. The (Cauchy) stress tensor components of the Ogden hyperelastic model under plane stress/strain and axisymmetric states are evaluated in subroutine SUOGD.

ARGUMENT LIST

Type	Name		Description
double precision	AMATX	<	Matrix of components of the spatial elasticity (fourth order) tensor.
double precision	B	>	Array of components of the left Cauchy-Green strain tensor.
integer	IPROPS	>	Array of integer material properties.
integer	NTYPE	>	Stress state type flag.
double precision	RPROPS	>	Array of real material properties.
double precision	STRES	>	Array of Cauchy stress tensor components.

SUBOUTINE CSTPDS - MANUAL PAGE

DESCRIPTION
Spatial tangent modulus for planar double slip single crystal model

This routine computes the (elastic or elasto-plastic) spatial tangent modulus, a, for the planar double slip single crystal elasto-plastic model.
The tangent modulus computed here is fully consistent with the exponential map-based elastic predictor/return mapping integration algorithm implemented in subroutine SUPDSC.
This model is restricted to the plane strain case.
The elastic behaviour of the single crystal is assumed isotropic (regularised) Neo-Hookean.

ARGUMENT LIST

Type	Name		Description
double precision	AMATX	<	Matrix of components of the spatial tangent (fourth order) modulus.
double precision	DGAM	>	Array of incremental plastic multipliers last determined in SUPDSC. All multipliers are set to zero for the first iteration of each load increment.
logical	EPFLAG	>	Elasto-plastic flag. If .FALSE., AMATX returns as the elastic modulus. If .TRUE., AMATX returns as the elasto-plastic modulus.
double precision	FINCR	>	Last incremental deformation gradient.
integer	IPROPS	>	Array of integer material properties.
logical	LALGVA	>	Array of logical algorithmic flags.
integer	NTYPE	>	Stress state type flag. The present implementation is compatible only with plane strain (NTYPE=2).
double precision	RPROPS	>	Array of real material properties.
double precision	RSTAVA	>	Array of current real state variables other than the stress tensor components.
double precision	RSTAVN	>	Array of real state variables other than the stress tensor components at the beginning of the curent load increment.
double precision	STRES	>	Array of current Cauchy stress tensor components.

SUBOUTINE CTDAMA - MANUAL PAGE

DESCRIPTION
Computation of consistent tangent matrix Lemaitre's damage model.

This routine computes the tangent matrix consistent with the fully implicit elastic predictor/return mapping algorithm for Lemaitre's ductile damage elasto-plastic model coded in subroutine SUDAMA.
It contains the plane strain and axisymmetric implementations.
It returns either the elastic tangent or the elasto-plastic consistent tangent matrix depending on the input value of EPFLAG.

ARGUMENT LIST

Type	Name		Description
double precision	DGAMA	>	Incremental plastic multiplier.
double precision	DMATX	<	Consistent tangent matrix.
logical	EPFLAG	>	Elasto-plastic flag. If .FALSE., DMATX returns as the elastic matrix. If .TRUE., DMATX returns as the elasto-plastic tangent consistent with the return mapping algorithm implemented in routine SUDAMA.
integer	IPROPS	>	Array of integer material properties. This array is set in routines INDATA RDDAMA. The number of points on the piece-wise linear hardening curve is the only element of this array used here.
integer	NTYPE	>	Stress state type flag.
double precision	RPROPS	>	Array of real material properties.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components. Current values. The state variables stored in this array are the (engineering) elastic strain components, the hardening and damage internal variables.
double precision	STRES	>	Array of current stress tensor components.

SUBOUTINE CTDMEL - MANUAL PAGE

DESCRIPTION
Computation of the tangent matrix for the damaged elastic model.

This routine returns the tangent matrix of the isotropically damaged isotropic elastic model with microcrack/void closure effects. It contains the plane strain axisymmetric implementations of the model.

ARGUMENT LIST

Type	Name		Description
double precision	DMATX	<	Tangent matrix.
integer	NTYPE	>	Stress state type flag.
double precision	RPROPS	>	Array of real material properties.
double precision	RSTAVA	>	Array of real state variables other than the stresses.
double precision	STRES	>	Array of stress components.

SUBOUTINE CTDP - MANUAL PAGE

DESCRIPTION
Consistent tangent matrix for the Drucker-Prager model

The tangent matrix computed in this routine is consistent with fully implicit elastic predictor/return mapping algorithm for the Drucker-Prager elasto-plastic material model with piece-wise linear isotropic hardening carried out in subroutine SUDP.
It contains only the plane strain and axisymmetric implementations.
It returns either the elastic tangent or the elasto-plastic consistent tangent matrix depending on the input value of the logical argument EPFLAG.

ARGUMENT LIST

Type	Name		Description
double precision	DGAM	>	Incremental plastic multipliers obtained in routine SUDP.
double precision	DMATX	<	Consistent tangent matrix.
logical	EPFLAG	>	Elasto-plastic flag. If .FALSE., DMATX returns as the elastic matrix. If .TRUE., DMATX returns as the elasto-plastic tangent consistent with the return mapping algorithm implemented in routine SUDP.
integer	IPROPS	>	Array of integer material properties. This array is set in routines INDATA and RDDP. The number of points on the piece-wise linear hardening curve is the only element of this array used here.
logical	LALGVA	>	Array of logical algorithmic flags. See list of arguments of SUDP.
integer	NTYPE	>	Stress state type.
double precision	RPROPS	>	Array of real material properties. Same as in the argument list of subroutine SUDP.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components. Output of SUDP.
double precision	STRAT	>	Array of elastic trial (engineering) strain components last used as input of subroutine SUDP.

SUBOUTINE CTDPPN - MANUAL PAGE

DESCRIPTION
Consistent tangent for the Drucker-Prager model in plane stress.

This routine computes the tangent matrix consistent with the nested iteration algorithm for the Drucker-Prager model in plane stress coded in subroutine SUDPPN.
It returns either the elastic tangent or the elasto-plastic consistent tangent matrix depending on the input value of the logical argument EPFLAG.

ARGUMENT LIST

Type	Name		Description
double precision	RALGVA	>	Array of real algorithmic variables. For the present plane stress implementation, it contains the incremental plastic multipliers obtained in routine SUDPPN and the elastic trial thickness strain obtained as the solution of the plane stress enforcement loop of SUDPPN. Note that for the first iteration of a load increment the incremental plastic multipliers must be set to zero and elastic trial thickness strain must be set equal to the elastic thickness strain of the previous equilibrium solution.
double precision	DMATX	<	Plane stress consistent tangent matrix.
logical	EPFLAG	>	Elasto-plastic flag. If .FALSE., DMATX returns as the elastic matrix (the standard plane stress linear elasticity matrix). If .TRUE., DMATX returns as the as the elasto-plastic tangent consistent with the nested iteration algorithm implemented in routine SUDPPN.
integer	IPROPS	>	Array of integer material properties. This array is set in routines INDATA and RDDP. The number of points on the piece-wise linear hardening curve is the only element of this array used here.
logical	LALGVA	>	Array of logical algorithmic flags. See list of arguments of SUDP.
integer	NTYPE	>	Stress state type flag.
double precision	RPROPS	>	Array of real material properties. Same as in the argument list of subroutines SUDP or SUDPPN.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components. Output of SUDP or SUDPPN.
double precision	STRAT	>	Array of elatic trial engineering strains.

SUBOUTINE CTEL - MANUAL PAGE

DESCRIPTION
Computation of the elastic matrix for the linear elastic model.

This routine returns the standard elasticity matrix of the linear elastic material model. It contains the plane stress, plane strain and axisymmetric implementations.

ARGUMENT LIST

Type	Name		Description
double precision	DMATX	<	Elastic matrix.
integer	NTYPE	>	Stress state type flag.
double precision	RPROPS	>	Array of real material properties.

SUBOUTINE CTMC - MANUAL PAGE

DESCRIPTION
Consistent tangent matrix for the Mohr-Coulomb model

The tangent matrix computed in this routine is consistent with fully implicit elastic predictor/return mapping algorithm for the Mohr-Coulomb elasto-plastic material model with piece-wise linear isotropic hardening carried out in subroutine SUMC.
It contains the plane strain and axisymmetric implementations of the model.
It returns either the elastic tangent or the elasto-plastic consistent tangent matrix depending on the input value of EPFLAG.

ARGUMENT LIST

Type	Name		Description
double precision	DMATX	<	Consistent tangent matrix.
logical	EPFLAG	>	Elasto-plastic flag. If .FALSE., DMATX returns as the elastic matrix. If .TRUE., DMATX returns as the elasto-plastic tangent consistent with the return mapping algorithm implemented in routine SUMC.
integer	IPROPS	>	Array of integer material properties. This array is set in routines INDATA and RDMC. The number of points on the piece-wise linear hardening curve is the only element of this array used here.
logical	LALGVA	>	Array of logical algorithmic flags. See list of arguments of SUMC.
integer	NTYPE	>	Stress state type.
double precision	RPROPS	>	Array of real material properties. Same as in the argument list of subroutine SUMC.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components. Output of SUMC.
double precision	STRAT	>	Array of elastic trial (engineering) strain components. Same as in the input SUMC.
double precision	STRES	>	Array of current stress tensor components.

SUBOUTINE CTTR - MANUAL PAGE

DESCRIPTION
Consistent tangent matrix for the Tresca model

The tangent matrix computed in this routine is consistent with fully implicit elastic predictor/return mapping algorithm for the Tresca elasto-plastic material model with piece-wise linear isotropic hardening carried out in subroutine SUTR.
This routine contains the plane strain and axisymmetric implementations of the model.
It returns either the elastic tangent or the elasto-plastic consistent tangent matrix depending on the input value of the logical argument EPFLAG.
The present implementation is based on the use of closed formulae for the derivatives of general isotropic tensor functions of one tensor in which the eigenvalues of the function are expressed as functions (the algorithmic incremental principal stress-based constitutive relation in the present case) of the eigenvalues of its tensor argument.

ARGUMENT LIST

Type	Name		Description
double precision	DMATX	<	Consistent tangent matrix.
logical	EPFLAG	>	Elasto-plastic flag. If .FALSE., DMATX returns as the elastic matrix. If .TRUE., DMATX returns as the elasto-plastic tangent consistent with the return mapping algorithm implemented in routine SUTR.
integer	IPROPS	>	Array of integer material properties. This array is set in routines INDATA and RDTR. The number of points on the piece-wise linear hardening curve is the only element of this array used here.
logical	LALGVA	>	Array of logical algorithmic flags. See list of arguments of SUTR.
integer	NTYPE	>	Stress state type.
double precision	RPROPS	>	Array of real material properties. Same as in the argument list of subroutine SUTR.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components. Output of SUTR.
double precision	STRAT	>	Array of elastic trial (engineering) strain components. Same as in the input of SUTR.
double precision	STRES	>	Array of current stress tensor components.

SUBOUTINE CTTRPN - MANUAL PAGE

DESCRIPTION
Consistent tangent matrix for the Tresca model in plane stress.

The tangent matrix computed in this routine is consistent with the nested iteration algorithm for the Tresca model in plane stress coded in subroutine SUTRPN.
It returns either the elastic tangent or the elasto-plastic consistent tangent matrix depending on the input value of the logical argument EPFLAG.

ARGUMENT LIST

Type	Name		Description
double precision	DMATX	<	Consistent tangent matrix.
logical	EPFLAG	>	Elasto-plastic flag. If .FALSE., DMATX returns as the elastic as the elastic matrix. If .TRUE., DMATX returns as the elasto-plastic tangent consistent with the return mapping algorithm implemented in routine SUTR.
integer	IPROPS	>	Array of integer material properties. This array is set in routines INDATA and RDTR. The number of points on the piece-wise linear hardening curve is the only element of this array used here.
logical	LALGVA	>	Array of logical algorithmic flags. See list of arguments of SUTR.
double precision	RPROPS	>	Array of real material properties. Same as in the argument list of subroutine SUTR.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components. Output of SUTR.
double precision	STRAT	>	Array of elastic trial (engineering) strain components. Same as in the input SUTR.
double precision	STRES	>	Array of current stress tensor components.

SUBOUTINE CTVM - MANUAL PAGE

DESCRIPTION
Computation of consistent tangent matrix for von Mises model.

This routine computes the tangent matrix consistent with the fully implicit elastic predictor/return mapping algorithm for the von Mises model coded in subroutine SUVM.
It contains the plane strain and axisymmetric implementations.
It returns either the elastic tangent or the elasto-plastic consistent tangent matrix depending on the input value of EPFLAG.

ARGUMENT LIST

Type	Name		Description
double precision	DGAMA	>	Incremental plastic multiplier.
double precision	DMATX	<	Consistent tangent matrix.
logical	EPFLAG	>	Elasto-plastic flag. If .FALSE., DMATX returns as the elastic matrix. If .TRUE., DMATX returns as the elasto-plastic tangent consistent with the return mapping algorithm implemented in routine SUVM.
integer	IPROPS	>	Array of integer material properties. This array is set in routines INDATA and RDVM. The number of points on the piece-wise linear hardening curve is the only element of this array used here.
integer	NTYPE	>	Stress state type flag.
double precision	RPROPS	>	Array of real material properties. This array contains the density (not used in this routine), the elastic properties: Young's modulus and Poisson's ratio, and the pairs ``accumulated plastic strain-uniaxial yield stress'' defining the (user supplied) piece-wise linear hardening curve. This array is set in routine RDVM.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components. Current values. The state variables stored in this array are the (engineering) elastic strain components and the accumulated plastic strain.
double precision	STRES	>	Array of current stress tensor components.

SUBOUTINE CTVMMX - MANUAL PAGE

DESCRIPTION
Consistent tangent matrix for von Mises model with mixed hardening.

This routine computes the tangent matrix consistent with the fully implicit elastic predictor/return mapping algorithm for the von Mises model with mixed hardening coded in subroutine SUVMMX.
It contains the plane strain and axisymmetric implementations.
It returns either the elastic tangent or the elasto-plastic consistent tangent matrix depending on the input value of the argument EPFLAG.

ARGUMENT LIST

Type	Name		Description
double precision	DGAMA	>	Incremental plastic multiplier.
double precision	DMATX	<	Consistent tangent matrix.
logical	EPFLAG	>	Elasto-plastic flag. If .FALSE., DMATX returns as the elastic matrix. If .TRUE., DMATX returns as the elasto-plastic tangent consistent with the return mapping algorithm implemented in routine SUVM.
integer	IPROPS	>	Array of integer material properties. This array is set in routines MATIRD and RDVM. The number of points on the piece-wise linear hardening curve is the only element of this array used here.
integer	NTYPE	>	Stress state type flag.
double precision	RPROPS	>	Array of real material properties. This array contains the density (not used in this routine), the elastic properties: Young's modulus and Poisson's ratio, the pairs ``accumulated plastic strain-isotropic hard. stress'' defining the isotropic hardening curve and the pairs ``accumulated plastic strain-kinematic hardening stress'' defining the kinematic hardening curve. This array is set in routine RDVMMX.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components. Current values. The state variables stored in this array are the (engineering) elastic strain components, the accumulated plastic strain and the backstress tensor components.
double precision	RSTAV2	>	Array of real state variables other than the stress tensor components. Values at last converged (equilibrium) solution.
double precision	STRES	>	Array of current stress tensor components.

SUBOUTINE CTVMPS - MANUAL PAGE

DESCRIPTION
Consistent tangent matrix for von Mises model in plane stress.

This routine computes the tangent matrix consistent with the plane stress-projected implicit elastic predictor/return mapping algorithm for the von Mises model coded in subroutine SUVMPS. It returns either the elastic tangent or the elasto-plastic consistent tangent matrix depending on the input value of the logical argument EPFLAG.

ARGUMENT LIST

Type	Name		Description
double precision	DGAMA	>	Incremental plastic multiplier (outcome of SUVMPS.
double precision	DMATX	<	Plane stress consistent tangent matrix.
logical	EPFLAG	>	Elasto-plastic flag. If .FALSE., DMATX returns as the elastic matrix (the standard plane stress linear elasticity matrix). If .TRUE., DMATX returns as the elasto-plastic tangent consistent with the plane stress-projected return mapping algorithm implemented in routine SUVMPS.
integer	IPROPS	>	Array of integer material properties. This array is set in routines INDATA and RDVM. The number of points on the piece-wise linear hardening curve is the only element of this array used here.
integer	NTYPE	>	Stress state type flag.
double precision	RPROPS	>	Array of real material properties. This array contains the density (not used in this routine), the elastic properties: Young's modulus and Poisson's ratio, and the pairs ``accumulated plastic strain-uniaxial yield stress'' defining the (user supplied) piece-wise linear hardening curve. This array is set in routine RDVM.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components. Current values (outcome of SUVMPS). The state variables stored in this array are the (engineering) elastic strain components and the accumulated plastic strain.
double precision	STRES	>	Array of current stress tensor components (outcome of SUVMPS).

DOUBLE PRECISION FUNCTION DDLGD2 - MANUAL PAGE

DESCRIPTION
ddlg2(x)=1/(2x)

This is the derivative of the function defined in DLGD2, that relates the principal logarithmic stretches and the eigenvalues of the Cauchy-Green strain tensor.

ARGUMENT LIST

Type	Name		Description
double precision	X	>	Point at which the function will be evaluated.

SUBOUTINE DEFGRA - MANUAL PAGE

DESCRIPTION
Deformation gradient for 2D isoparametric finite element

Given the element nodal displacements and the discrete gradient operator, G-matrix, at a point, this routine computes the corresponding deformation gradient at that point. This routine contains the plane strain, plane stress and axisymmetric implementations.

ARGUMENT LIST

Type	Name		Description
double precision	ELDISP	>	Array of nodal displacements of the finite element.
double precision	F	<	Deformation gradient.
double precision	GMATX	>	Discrete (full) gradient operator, G-matrix, at the point of interest.
integer	MDOFN	>	Dimensioning parameter: number of rows of array ELDISP.
integer	MGDIM	>	Dimensioning parameter: number of rows of array GMATX.
integer	NDOFN	>	Number of degrees of freedom per node.
integer	NTYPE	>	Stress state type flag.
integer	NNODE	>	Number of nodes of the element.

SUBOUTINE DEXPMP - MANUAL PAGE

DESCRIPTION
Derivative of exponential map for general three-dimensional tensors

This routine computes the derivative of the exponential of a generally unsymmetric three-dimensional tensor.
It uses the series definition of the tensor exponential.
The exponential map itself is implemented in subroutine EXPMAP.

ARGUMENT LIST

Type	Name		Description
double precision	DEXPX	<	Derivative of the exponential map at X. This derivative is a fourth order tensor stored here as a 4-index array.
logical	NOCONV	<	Logical convergence flag. Set to .TRUE. if the series fail to converge Set to .FALSE. otherwise.
double precision	X	>	Tensor at which exponential derivative is to be computed.

SUBOUTINE DGISO2 - MANUAL PAGE

DESCRIPTION
Derivative of a general isotropic tensor function of one tensor

This function computes the derivative, dY(X)/dX, of a general isotropic tensor function of one tensor, Y(X). This implementation is restricted to 2-D with one possible out-of-plane component (normally needed in axisymmetric problems). The tensor function Y(X) is assumed to be defined as Y(X)= Sum[yi(x1,x2,x3) ei(x)ei], where yi are the eigenvalues of the tensor Y and xi the eigenvalues of the tensor X. ei are the egenvectors of X (which by definition of Y(X), coincide with those of tensor Y) and "(x)" denotes the tensor product.

ARGUMENT LIST

Type	Name		Description
double precision	DEIGY	>	Matrix containing the derivatives dyi/dxj of the eigenvalues of Y(X) with respect to the eigenvalues of X.
double precision	DYDX	<	Matrix of components of the derivative (fourth order tensor) dY/dX.
double precision	EIGPRJ	>	Matrix with each column containing the components of one eigenprojection tensor ei(x)ei.
double precision	EIGX	>	Array of eigenvalues of X.
double precision	EIGY	>	Array of eigenvalues of Y.
logical	OUTOFP	>	Out-of-plane component flag. If set to .TRUE. the out-of-plane component (normally required in axisymmetric problems) is computed. The out-of-plane component is not computed otherwise.
logical	REPEAT	>	Repeated in-plane eigenvalues flag. If the in-plane eigenvalues are repeated this argument must be set to .TRUE. on entry, so that the appropriate limit expression for the the derivative is employed.

SUBOUTINE DISO2 - MANUAL PAGE

DESCRIPTION
Derivative of a class of isotropic tensor function of one tensor

This function computes the derivative, dY(X)/dX, of a particular
class of symmetric isotropic tensor valued function of one
symmetric tensor, Y(X).
This implementation is restricted to 2-D with one possible
out-of-plane component (normally needed in axisymmetric problems).
The class of tensor functions Y(X) is assumed to be defined as
Y(X)= Sum[y(xi) ei(x)ei], where the scalar function y(xi) defines
the eigenvalues of the tensor Y and xi the eigenvalues of the
tensor X. ei are the egenvectors of X (which by definition of
Y(X), coincide with those of tensor Y) and "(x)" denotes the
tensor product.

ARGUMENT LIST

Type	Name		Description
double precision	DYDX	<	Matrix of components of the derivative (fourth order tensor) dY/dX.
symbolic name	DFUNC	>	Symbolic name of the double precision function defining the derivative dy(x)/dx of the eigenvalues of the tensor function.
symbolic name	FUNC	>	Symbolic name of the double precision function defining the eigenvalues y(x) of the tensor fumction.
logical	OUTOFP	>	Out-of-plane component flag. If set to .TRUE. the out-of-plane component (normally required in axisymmetric problems) is computed. The out-of-plane component is not computed otherwise.
double precision	X	>	Point (argument) at which the derivative is to be computed.

DOUBLE PRECISION FUNCTION DLGD2 - MANUAL PAGE

DESCRIPTION

dlg2(x)=log(x)/2

This function relates the principal logarithmic stretches and the eigenvalues of the Cauchy-Green strain tensor.

ARGUMENT LIST

Type	Name		Description
double precision	X	>	Point at which the function will be evaluated.

DOUBLE PRECISION FUNCTION DPLFUN - MANUAL PAGE

DESCRIPTION

Returns the derivative of piece-wise linear scalar function

This procedure returns the derivative of the piece-wise linear scalar function of procedure PLFUN. The piece-wise linear function F(X) is defined by a set of NPOINT pairs (X,F(X)) passed in the matrix argument XFX.

ARGUMENT LIST

Type	Name		Description
double precision	X	>	Point at which the derivative will be evaluated.
integer	NPOINT	>	Number of points defining the piece-wise linear function.
double precision	XFX	>	Matrix (dimension 2*NPOINT) containing the pairs (x,f(x)) which define the piece-wise linear function. evaluated. Each column of XFX contains a pair (xi,f(xi)). The pairs supplied in XFX must be ordered such that the x's are monotonically increasing. That is, the x [XFX(1,i+1)] of a column i+1 must be greater than XFX(1,i) (x of column i). If X < XFX(1,1) the piece-wise linear function is assumed constant equal to XFX(1,1). If X > XFX(1,NPOINT) the piece-wise linear function is assumed constant equal to XFX(1,NPOINT).

SUBOUTINE ELEIIF - MANUAL PAGE

DESCRIPTION
Element interface for internal force vector calculation

This routine calls the internal force vector calculation routines for all element classes available in HYPLAS.
If the internal force vector calculation fails for some reason (such as due to failure of a material-specific state update procedure) the return value of the logical argument IFFAIL will be .TRUE., which will activate the increment cutting procedure in the main program.

ARGUMENT LIST

Type	Name		Description
integer	IELEM	>	Number of the element whose internal force vector is to be computed.
logical	IFFAIL	<	Internal force computation failure flag. Return value set to .FALSE. if internal force vector was successfully evaluated and set to .TRUE. otherwise.

SUBOUTINE ELEIST - MANUAL PAGE

DESCRIPTION
Element interface for element tangent stiffness matrix computation

This routine calls the tangent stiffness matrix calculation routines for all element classes available in HYPLAS.

ARGUMENT LIST

Type	Name		Description
double precision	ESTIF	<	Tangent stiffness matrix of the current element.
integer	IELEM	>	Number of the element whose stiffness matrix is to be computed.
integer	KUNLD	>	Unloading flag. KUNLD is set to 1 if the the loading programme is currently unloading.
logical	UNSYM	>	Tangent stiffness symmetry flag. .TRUE. if tangent stiffness is unsymmetric, .FALSE. otherwise.

SUBOUTINE ERRPRT - MANUAL PAGE

DESCRIPTION
Prints error message and can abort the program if requested

This routine prints error/warning messages to the standard output and results file and may abort the program depending on the entry value of its argument. The character string passed as its argument is an error code which will be searched for in the file ERROR.RUN, assumed to be kept in the directory defined by the HYPLASHOME environment variable. If the correponding character string (error code) is found, the associated error/warning message is printed in the standard output and results file.
The are 5 types of errors/warnings: Input data error, input data warning, internal error, execution error and execution warning. These are characterised, respectively, by error codes of the types: 'ED????', 'WD????', 'EI????', 'EE????' and 'WE????'.
This routine aborts the program in case of input data error (ED????), internal error ('EI????') or execution error (EE????).
The execution of the program is not interrupted in case of warnings ('WE????' or 'WD????').
See file ERROR.RUN for more details.

ARGUMENT LIST

Type	Name	Description
character	ERRCOD >	Character string containing the error code.

DOUBLE PRECISION FUNCTION EXP2X - MANUAL PAGE

DESCRIPTION

f(x)=exp(2x)

This function relates the eigenvalues of the Cauchy-Green strain tensors to the principal logarithmic stretches.

ARGUMENT LIST

Type	Name		Description
double precision	X	>	Point at which the function will be evaluated.

SUBOUTINE EXPMAP - MANUAL PAGE

DESCRIPTION
Exponential map for general three-dimensional tensors

This routine computes the exponential of a generally unsymmetric three-dimensional tensor. It uses the series definition of the tensor exponential

ARGUMENT LIST

Type	Name		Description
double precision	EXPX	<	Tensor exponential of X.
logical	NOCONV	<	Logical convergence flag. Set to .TRUE. if the series fail to converge. Set to .FALSE. otherwise.
double precision	X	>	Tensor whose exponential is to be computed.

SUBOUTINE EXQ4 - MANUAL PAGE

DESCRIPTION
Sets Gauss point-node extrapolation matrix for element type QUAD4

This routine sets the coefficients matrix for extrapolation of
fields from Gauss point values to nodal values for element type
QUAD4: Standard isoparametric 4-noded bi-linear quadrilateral.

ARGUMENT LIST

Type	Name	Description
integer	NGAUSP >	Number of Gauss points.
double precision	EXMATX <	Extrapolation matrix.

SUBOUTINE EXQ4FB - MANUAL PAGE

DESCRIPTION
Gauss point-node extrapolation matrix for element type QUA4FB

This routine sets the coefficients matrix for extrapolation of fields from Gauss point values to nodal values for element type QUA4FB: F-bar isoparametric 4-noded bi-linear quadrilateral.

ARGUMENT LIST

Type	Name	Description
double precision	EXMATX <	Extrapolation matrix.

SUBOUTINE EXQ8 - MANUAL PAGE

DESCRIPTION
Gauss point-node extrapolation matrix for element type QUAD8

This routine sets the coefficients matrix for extrapolation of fields from Gauss point values to nodal values for element type QUAD8: Standard isoparametric 8-noded quadrilateral.

ARGUMENT LIST

Type	Name	Description
integer	NGAUSP >	Number of Gauss points.
double precision	EXMATX <	Extrapolation matrix.

SUBOUTINE EXT3 - MANUAL PAGE

DESCRIPTION
Gauss point-node extrapolation matrix for element type TRI3

This routine sets the coefficients matrix for extrapolation of fields from Gauss point values to nodal values for element type TRI3: Standard isoparametric 3-noded linear triangle.

ARGUMENT LIST

Type	Name	Description
double precision	EXMATX <	Extrapolation matrix.

SUBOUTINE EXTNOD - MANUAL PAGE

DESCRIPTION
Extrapolates Gauss point values of a given field to nodes

This routine extrapolates the Gauss point values of a given field to the nodes of a finite element. It simply multiplies the coefficients matrix for extrapolation EXMATX by the matrix VARGP which contains the value of each component of the given field in all integration points of the element.
The results stored in the argument VARNOD are the extrapolated (or locally smoothed) values of the components of the given field at the nodes of the element.

ARGUMENT LIST

Type	Name		Description
double precision	EXMATX	>	Extrapolation matrix. Dimension NNODE x NGAUSP.
double precision	VARGP	>	Matrix containing, in each column, the Gauss point value of all components of the given field. Dimension NVAR x NGAUSP.
double precision	VARNOD	<	Matrix containing, in each column, the (extrapolated) nodal value of all components of the given field. Dimension NVAR x NNODE.
integer	NVAR	>	Number of components of the given field.
integer	NGAUSP	>	Number of Gauss points.
integer	NNODE	>	Number of nodes of the element.

SUBOUTINE FCLOSE - MANUAL PAGE

DESCRIPTION
Closes data file and results file

SUBOUTINE FNDKEY - MANUAL PAGE

DESCRIPTION
Finds and reads a line containing a specified keyword from a file.

This routine searches for a given keyword positioned as the first word of a line in a file.
If the given keyword is found then the corresponding line is read and returned together with the number of words in the line and two integer arrays containing the position of the beginning and end of each word.
If there is more than one line in the given file containing the the specified keyword as its first word, then this routine will return the first line found in the search which, in general, does not coincide with the first occurrence of the keyword in the file. In this case, which line is returned depends on which line of the file the search started. Hence, it is recommended that files to be searched contain only one occurrence of the keyword as first word of a line (occurrences of the keyword in positions other than as the first word of a line are not detected by this routine and can safely be present in the file).

ARGUMENT LIST

Type	Name		Description
logical	FOUND	<	Logical flag. Its return value is set to .TRUE. if the specified keyword is found at the beginning of a line in the specified file and set to .FALSE. otherwise.
integer	IWBEG	<	Integer array containg the position of the beginning of each word in the line containing the specified keyword. IWBEG(K) is the position of the beginning of the Kth word in the line.
integer	IWEND	<	Integer array containg the position of the end of each word in the line containing the specified keyword. IWEND(K) is the position of the end of the Kth word in the line.
character	KEYWRD	>	Keyword.
character	INLINE	<	Contents of the line whose first word is the specified keyword.
integer	NFILE	>	Unit identifier of the file to be searched.
integer	NWRD	<	Total number of words in the line containing the specified keyword.

SUBOUTINE FOPEN - MANUAL PAGE

DESCRIPTION
Opens/sets the names of the input data, results and re-start files

This routine reads the input data file name from the standard input, sets the names and opens the corresponding data and results files, and then sets the re-start file name.

ARGUMENT LIST

Type	Name	Description
character	DATFIL <	Character string containing the input data file name.
character	RESFIL <	Character string containing the results output file name.
character	RSTOUT <	Character string containing the re-start output file name.

SUBOUTINE FRONT - MANUAL PAGE

DESCRIPTION
Frontal solver

Assembles and solves the global system of linear algebraic finite element equilibrium equations by the frontal method. For non-linear problems, assembles and solves the corresponding linearised (or approximately linearised system of) equations.

ARGUMENT LIST

Type	Name		Description
integer	IITER	>	Current equilibrium iteration number.
integer	KRESL	>	Equation resolution index.
integer	IFNEG	<	Signum (+1/-1) of the determinant of the stiffness matrix.
integer	KUNLD	<>	Unloading flag.
double precision	MXFRON	>	Maximum frontwidth encountered in the system of linear finite element equations.
logical	UNSYM	>	Stiffness matrix unsymmetry flag.
logical	INCCUT	<	Load increment cutting flag.

SUBOUTINE GAUS1D - MANUAL PAGE

DESCRIPTION

Set Gauss point positions and weights for 1-D Gauss quadratures

Given the required number of integration points,
this routine sets the sampling point positions and the
corresponding weights for 1-D Gauss quadratures for numerical
integration over the domain [-1,1].

ARGUMENT LIST

Type	Name		Description
integer	NGAUS	>	Number of Gauss points.
double precision	POSGP	<	Array containing the Gauss point positions in the standard domain.
double precision	WEIGP	<	Array containing the Gauss point weights.

SUBOUTINE GAUS2D - MANUAL PAGE

DESCRIPTION
Sets Gaussian quadrature constants for 2-D domains

Given the type of domain and the required number of integration points, this routine sets the sampling point positions and the corresponding weights following standard Gauss quadrature rules for 2-D domains (quadrilateral and triangular domains).

ARGUMENT LIST

Type	Name		Description
character	DOMAIN	>	Character string with domain type flag. Entry value of DOMAIN can be either 'QUA' (for quadrilateral domains or 'TRI' (for triangular domains).
integer	NGAUS	>	Number of integration points.
double precision	POSGP	<	Array containing the position of the integration points.
double precision	WEIGP	<	Array containing the weights of the integration points.

SUBOUTINE GETBMX - MANUAL PAGE

DESCRIPTION
Computes the discrete symmetric gradient operator for 2-D elements

This routine assembles the discrete symmetric gradient operator (strain-displacement matrix in small strains), the B-matrix, for isoparametric 2-D finite elements: Plane strain, plane stress and axisymmetric cases.

ARGUMENT LIST

Type	Name		Description
double precision	BMATX	<	The discrete symmetric gradient operator, B-matrix.
double precision	CARTCO	>	Cartesian coordinates of the point where the B-matrix is to be computed.
double precision	CARTD	>	Array of cartesian derivatives of the element shape functions at the point of interest.
integer	NDIME	>	Dimensioning parameter: Number of rows of CARTCO and CARTD.
integer	MBDIM	>	Dimensioning parameter: Number of rows of BMATX.
integer	NAXIS	>	Axis of symmetry flag. Used only for the axisymmetric case.
integer	NNODE	>	Number of nodes of the element.
integer	NTYPE	>	Stress state type flag.
double precision	SHAPE	>	Array containing the value of the shape functions at the point of interest.

SUBOUTINE GETGCO - MANUAL PAGE

DESCRIPTION
Gets coordinates of a point within an element by interpolation

This routine computes the global cartesian coordinates of a point within a finite element by interpolation of its nodal coordinates.

ARGUMENT LIST

Type	Name		Description
double precision	CARTCO	<	Cartesian coordinates of the point of interest.
double precision	ELCOD	>	Array of nodal coordinates of the element.
integer	MDIME	>	Dimensioning parameter: Number of rows of ELCOD.
integer	NDIME	>	Number of spatial dimensions.
integer	NNODE	>	Number of nodes of the element.
double precision	SHAPE	>	Array containing the value of the shape function at the point of interest.

SUBOUTINE GETGMX - MANUAL PAGE

DESCRIPTION
Computes the discrete (full) gradient operator for 2-D elements

This routine assembles the discrete gradient operator, the G-matrix for isoparametric 2-D finite elements: Plane strain, plane stress and axisymmetric cases.

ARGUMENT LIST

Type	Name		Description
double precision	CARTCO	>	Cartesian coordinates of the point where the G-matrix is to be computed.
double precision	CARTD	>	Array of cartesian derivatives of the element shape functions at the point of interest.
double precision	GMATX	<	The discrete gradient operator, G-matrix.
integer	MDIME	>	Dimensioning parameter: Number of rows of CARTD.
integer	MGDIM	>	Dimensioning parameter: Number of rows of GMATX.
integer	NAXIS	>	Axis of symmetry flag. Used only for the axisymmetric case.
integer	NNODE	>	Number of nodes of the element.
integer	NTYPE	>	Stress state type flag.
double precision	SHAPE	>	Array containing the value of the shape functions at the point of interest.

SUBOUTINE GREET - MANUAL PAGE

DESCRIPTION
Prints HYPLAS greeting message on the standard output

PROGRAM HYPLAS - MANUAL PAGE

DESCRIPTION
This is HYPLAS main program

SUBOUTINE IFFBA2 - MANUAL PAGE

DESCRIPTION

Internal force vector computation for 2-D F-bar elements

This routine computes the element internal force vector for all 2-D elements of the class FBAR: F-bar elements for plane strain and axisymmetric analysis under large deformations.

ARGUMENT LIST

Type	Name		Description
integer	IELEM	>	Number of the element whose internal force vector is to be computed.
logical	INCCUT	<	Increment cutting flag. Return value set to .FALSE. if internal force vector was successfully evaluated and set to .TRUE. otherwise. When the return value is set to .TRUE., the main program will activate increment cutting and the current increment will be divided into sub-increments.
integer	MDIME	>	Maximum permissible number of spatial dimensions.
integer	MELEM	>	Maximum permissible number of elements in a mesh.
integer	MPOIN	>	Maximum permissible number of nodal points in a mesh.
integer	MSTRE	>	Maximum permissible number of stress (or resultant force) components.
integer	MTOTV	>	Maximum permissible degrees of freedom in a mesh.
integer	NAXIS	>	Symmetry axis flag. Used only for axisymmetric analysis. NAXIS=1 if symmetric about the X-axis and NAXIS=2 if symmetric about the Y-axis.
integer	NTYPE	>	Stress state type flag.
double precision	COORD1	>	Array of coordinates of all nodes of the mesh in the current configuration.
double precision	DINCR	>	Global array of incremental displacements of the entire mesh.
double precision	ELOAD	<	Internal force vector of the current element.
integer	IELPRP	>	Array of integer element properties of the current element group.
integer	IPROPS	>	Array of integer material properties of the current element group.
logical	LALGVA	<	Array of current logical algorithmic variables of all Gauss points of the current element.
integer	LNODS	>	Global array of connectivity of the entire mesh.
double precision	RALGVA	<	Array of current real algorithmic variables of all Gauss points of the current element.
double precision	RELPRP	>	Array of real element properties of the current element group.
double precision	RPROPS	>	Array of real material properties of the current element group.
double precision	RSTAVA	<	Array of current real state variables of all Gauss points of the current element.
double precision	STRSG	<	Array of resultant forces of all centroids line Gauss points of the current element.

double precision	THKGP	<	Array of current thickness of all Gauss points of the current element. Used only in plane stress analysis. Updated here only in large strain analysis.
double precision	TDISP	>	Global array of total displacement of all nodes of the mesh.

SUBOUTINE IFSTD2 - MANUAL PAGE

DESCRIPTION
Internal force vector computation for 2-D elements of class STDARD

This routine computes the element internal force vector for all
2-D elements of the class STDARD: standard isoparametric elements
for plane strain, plane stress and axisymmetric analysis.

ARGUMENT LIST

Type	Name		Description
integer	IELEM	>	Number of the element whose internal force vector is to be computed.
logical	INCCUT	<	Increment cutting flag. Return value set to .FALSE. if internal force vector was successfully evaluated and set to .TRUE. otherwise. When the return value is set to .TRUE., the main program will activate increment cutting and the current increment will be divided into sub-increments.
integer	MDIME	>	Maximum permissible number of spatial dimensions.
integer	MELEM	>	Maximum permissible number of elements in a mesh.
integer	MPOIN	>	Maximum permissible number of nodal points in a mesh.
integer	MSTRE	>	Maximum permissible number of stress (or resultant force) components.
integer	MTOTV	>	Maximum permissible degrees of freedom in a mesh.
integer	NAXIS	>	Symmetry axis flag. Used only for axisymmetric analysis. NAXIS=1 if symmetric about the X-axis and NAXIS=2 if symmetric about Y-axis.
integer	NLARGE	>	Large deformation flag. Large deformation analysis if NLARGE=1 and infinitesimal deformation analysis otherwise.
integer	NTYPE	>	Stress state type flag.
double precision	COORD1	>	Array of coordinates of all nodes of the mesh in the current configuration.
double precision	DINCR	>	Global array of incremental displacements of the entire mesh.
double precision	ELOAD	<	Internal force vector of the current element.
integer	IELPRP	>	Array of integer element properties of the current element group.
integer	IPROPS	>	Array of integer material properties of the current element group.
logical	LALGVA	<	Array of current logical algorithmic variables of all Gauss points of the current element.
integer	LNODS	>	Global array of connectivity of the entire mesh.
double precision	RALGVA	<	Array of current real algorithmic variables of all Gauss points of the current element.
double precision	RELPRP	>	Array of real element properties of the current element group.
double precision	RPROPS	>	Array of real material properties of the current element group.
double precision	RSTAVA	<	Array of current real state variables of all Gauss points of the

```
                                   current element.

double precision   STRSG  <  Array of resultant forces of all
                             centroids line Gauss points of the
                             current element.

double precision   THKGP  <  Array of current thickness of
                             all Gauss points of the current element.
                             Used only in plane stress analysis.
                             Updated here only in large strain
                             analysis.

double precision   TDISP  >  Global array of total displacement
                             of all nodes of the mesh.
```

SUBOUTINE INCREM - MANUAL PAGE

DESCRIPTION
Increments the applied external load for fixed load increments.

This routine increments the applied external (proportional) load, sets the global unloading flag KUNLD and outputs the current load increment parameters to the results output file, when HYPLAS is running under the fixed load increments option.

ARGUMENT LIST

Type	Name		Description
integer	IINCS	>	Current load increment number.
double precision	TFACT	<>	Total load factor.
double precision	TOLER	<>	Equilibrium convergence tolerance.
integer	MITER	>	Maximum permissible number of equilibrium iterations.
integer	NOUTP	>	Printing output flag.
double precision	DFACT	>	Incremental load factor.
double precision	DFOLD	<	Stores the incremental load factor of previous load step.
integer	KUNLD	<	Unloading flag.

SUBOUTINE INDATA - MANUAL PAGE

DESCRIPTION
Reads most of the input data from a data file

ARGUMENT LIST

Type	Name		Description
integer	MXFRON	<	Maximum front size encountered in the specified mesh.
logical	UNSYM	<	Global unsymmetric tangent stiffness flag. Set to .TRUE. if any element and/or material produces an unsymmetric tangent stiffness requiring the unsymmetric solver during equilibrium iterations.

SUBOUTINE ININCR - MANUAL PAGE

DESCRIPTION
Reads input data for load increments

This routine reads from the data file and echos to the results file the data required for load incrementation.

ARGUMENT LIST

Type	Name		Description
double precision	DFACT	<	Initial incremental load factor (for Arc-Length control).
double precision	DLENP	<	Maximum arc-length parameter (for Arc-Length control).
double precision	FSTOP	<	Maximum total load factor, above which the analysis will stop (for Arc-Length control).
integer	ITDES	<	Desired number of equilibrium iterations (for Arc-Length control).
integer	MINCS	>	Dimensioning parameter: maximum permissible number of prescribed load increments under fixed load increments option. Defines size of DFACTV, MITERV and NOUTPV.
integer	MITER	<	Maximum allowed number of equilibrium iterations in any increment (for Arc-Length control).
integer	NALGO	>	Solution algorithm flag (positive for fixed increments option, negative for Arc-Length control).
integer	NINCS	<	For fixed increments option: total number of specified load increments. For Arc-Length control: maximum specified number of load increments.
double precision	TOLER	<	Equilibrium convergence for all load increments (for Arc-Length control).
double precision	DFACTV	<	Array with the incremental load factor of each specified load increment (for fixed load increments option).
integer	MITERV	<	Array with the maximum allowed number of global (equilibrium) iterations for each specified load increment (for fixed load increments option).
integer	NOUTP	<	Output code array with output frequency flags (for Arc-Length control).
integer	NOUTPV	<	Array with the output code for each specified load increment (for fixed load increments option).
double precision	TOLERV	<	Array with the equilibrium convergence tolerance for each specified load increment (for fixed load increments option).

SUBOUTINE INITIA - MANUAL PAGE

DESCRIPTION

Initialises some arrays and problem control variables.

This routine initialises various arrays and problem control variables. Gauss-point-related quantities (such as stresses and other state and algorithmic variables) are initialised by calling the corresponding material interface routine.

ARGUMENT LIST

Type	Name		Description
double precision	DLAMD	<	Iterative load factor.
integer	IFNEG	<	Signum (+1/-1) of the stiffness matrix determinant.
integer	KLUND	<	Unloading flag.
double precision	TFACT	<	Total load factor.

SUBOUTINE INLOAD - MANUAL PAGE

DESCRIPTION
Reads external loads and assembles the external force vector.

This routine reads the prescribed body forces, surface tractions and point (nodal) loads from the input data file and assembles the corresponding global external force vector.

SUBOUTINE INTFOR - MANUAL PAGE

DESCRIPTION
Calls internal force vector calculation routines

This routine calls the internal force vector calculation routines for all element classes available in HYPLAS. It loops over all elements of the mesh. If the internal force vector calculation fails for some reason (such as due to failure of an state update procedure) the return value of the logical argument INCCUT will be .TRUE., which will activate the increment cutting procedure in the main program.

ARGUMENT LIST

Type	Name	Description
logical	INCCUT <	Increment cutting flag. Return value set to .FALSE. if internal force vector was successfully evaluated and set to .TRUE. otherwise. When the return value is set to .TRUE., the main program will activate increment cutting and the current increment will be divided into sub-increments.

INTEGER FUNCTION INTNUM - MANUAL PAGE

DESCRIPTION

Converts a character string into an integer

This function returns the integer corresponding to the number contained in the character string passed as argument.

ARGUMENT LIST

Type	Name	Description
character	CHRSTR >	Character string containing a number.

SUBOUTINE INVF2 - MANUAL PAGE

DESCRIPTION
Inverts the deformation gradient for 2-D problems

This routine inverts deformation gradient tensors for plane strain,
plane stress and axisymmetric problems.

ARGUMENT LIST

Type	Name		Description
double precision	F	>	Deformation gradient.
double precision	FINV	<	Inverse of the deformation gradient.
integer	NTYPE	>	Stress state type flag.

SUBOUTINE INVMT3 - MANUAL PAGE

DESCRIPTION
Inverts a 3x3 double precision matrix

This routine inverts a generally unsymmetric 3x3 double precision matrix.

ARGUMENT LIST

Type	Name		Description
double precision	S	>	Matrix to be inverted
double precision	SINV	<	Inverse matrix
double precision	DETS	<	Determinant of matrix S

SUBOUTINE ISO2 - MANUAL PAGE

DESCRIPTION
Computes the value of isotropic tensor functions of one tensor.

This subroutine evaluates isotropic tensor functions Y(X), of one tensor belonging to the class described below.
This implementation is restricted to 2-D with one possible out-of-plane component (normally needed in axisymmetric problems).
The class of symmetric tensor functions Y(X) is assumed to be defined as Y(X)= Sum[y(xi) ei(x)ei], where the scalar function y(xi) defines the eigenvalues of the tensor Y and xi the eigenvalues of the tensor X. ei are the egenvectors of X (which by definition of Y(X), coincide with those of tensor Y) and "(x)" denotes the tensor product.

ARGUMENT LIST

Type	Name		Description
symbolic name	FUNC	>	Symbolic name of the double precision function defining y(xi).
logical	OUTOFP	>	Out-of-plane component flag. If set to .TRUE. the out-of-plane component (normally required in axisymmetric problems) is computed. The out-of-plane component is not computed otherwise.
double precision	X	>	Array of components of the tensor at which the function is to be evaluated.
double precision	Y	<	Array of components of the tensor function at X.

SUBOUTINE IVZERO - MANUAL PAGE

DESCRIPTION
Zero an integer array

This routine initialises to zero the N components of the integer array argument IV.

ARGUMENT LIST

Type	Name		Description
integer	IV	<	Zeroed integer array.
integer	N	>	Dimension of IV.

SUBOUTINE JACOB - MANUAL PAGE

DESCRIPTION

Jacobi procedure for spectral decomposition of a symmetric matrix

This routine uses the Jacobi iterative procedure for the spectral decomposition (decomposition into eigenvalues and eigenvectors) of a symmetric matrix.

ARGUMENT LIST

Type	Name		Description
double precision	A	<>	Matrix to be decomposed.
double precision	D	<	Array containing eigenvalues of A.
double precision	V	<	Matrix containing one eigenvector of A in each column.
integer	N	>	Dimension of A.

SUBOUTINE JACOB2 - MANUAL PAGE

DESCRIPTION
Jacobian det. and cartesian deriv. for 2-D isoparametric elements.

This routine computes the Jacobian of the isoparametric mapping for 2-D isoparametric elements and returns the Jacobian determinant and the cartesian derivatives of the shape functions.

ARGUMENT LIST

Type	Name		Description
CARTD			
double precision	CARTD	<	Matrix of cartesian derivatives of the element shape functions.
double precision	DERIV	>	Matrix of isoparametric derivatives of the element shape functions.
double precision	DETJAC	<	Isoparametric mapping Jacobian determinant.
double precision	ELCOD	>	Array of element nodal coordinates.
integer	IELEM	>	Element number (used only if warning messages are issued).
integer	MDIME	>	Dimensioning parameter for arguments CARTD, DERIV and ELCOD.
integer	NDIME	>	Number of spatial dimensions.
integer	NNODE	>	Number of nodes of the element.

SUBOUTINE LEFTCG - MANUAL PAGE

DESCRIPTION
Computes the left Cauchy-Green strain tensor

Given the previous total left Cauchy-Green strain tensor and the incremental deformation gradient between the previous and current configuration, this routine computes the current left Cauchy-Green strain tensor. This routine contains the plane strain, plane stress and axisymmetric implementations.

ARGUMENT LIST

Type	Name		Description
double precision	BN	>	Array of components of the previous (at tn) left Cauchy-Green tensor.
double precision	BNP1	<	Array of components of the current (at tn+1) left Cauchy-Green strain tensor.
double precision	FINCR	>	Incremental deformation gradient between the previous and current configuration.
integer	NTYPE	>	Stress state type. Present routine is compatible with NTYPE=1 (plane stress), NTYPE=2 (plane strain) and NTYPE=3 (axisymmetric condition).

SUBOUTINE LENGTH - MANUAL PAGE

DESCRIPTION
Adjusts step length for the Arc-Length Method.

This subroutine adjusts the step length for the Arc-Length Method according to the prescribed desired number of iterations for equilibrium convergence and the actual number of iterations for convergence in the previous load step.

ARGUMENT LIST

Type	Name		Description
double precision	DLEN	<	Calculated step length.
double precision	DLENM	>	Prescribed maximum permissible step length.
integer	ITACT	>	Actual number of iterations required for convergence in the previous load step.
integer	ITDES	>	Desired number of iterations for convergence in the iterative solution of the non-linear finite element equilibrium equations.

SUBOUTINE LISTRA - MANUAL PAGE

DESCRIPTION
Computes the infinitesimal strain components in 2-D

Given the nodal displacements of the element and the B-matrix (discrete symmetric gradient operator) at a point in the element domain, this routine computes the corresponding (engineering) infinitesimal strain components at that point by performing the standard operation: e = B u, where e is the array of engineering strain components and u is the array of nodal displacements. This routine contains the plane strain/stress and axisymmetric implementations.

ARGUMENT LIST

Type	Name		Description
double precision	BMATX	>	The discrete symmetric gradient operator, B-matrix.
double precision	ELDISP	>	Array containing the element nodal displacements.
integer	MDOFN	>	Dimensioning parameter: Number of rows of ELDISP.
integer	MBDIM	>	Dimensioning parameter: Number of rows of BMATX.
integer	NDOFN	>	Number of degrees of freedom per node.
integer	NNODE	>	Number of nodes of the element.
integer	NTYPE	>	Stress state type flag.
double precision	STRAN	<	Array of engineering infinitesimal strain components.

SUBOUTINE LOGSTR - MANUAL PAGE

DESCRIPTION
Logarithmic strain computation

Given the left (right) Cauchy-Green strain tensor, this routine computes the corresponding Eulerian (Lagrangian) logarithmic strain tensor (engineering components).
Plane strain, plane stress and axisymmetric implementations.

ARGUMENT LIST

Type	Name		Description
double precision	B	>	Array of components of the Cauchy-Green strain tensor.
double precision	E	<	Array of (engineering) components of the logarithmic strain tensor.
integer	NTYPE	>	Stress state type flag.

SUBOUTINE MATICT - MANUAL PAGE

DESCRIPTION
Material interface for consistent tangent routine calls

This routine calls the consistent tangent computation routine according to the material type. Given the necessary material-independent kinematic quantities computed at the element level (and passed into the present routine through its list of arguments) this routine identifies the material type in question and calls the corresponding material-specific tangent computation routine.

ARGUMENT LIST

Type	Name		Description
double precision	DETF	>	Current total deformation gradient at the Gauss point of interest.
integer	IITER	>	Number of the current equilibrium iteration.
integer	KUNLD	>	Unloading flag. KUNLD is set to 1 if the loading programme is currently unloading.
integer	MBDIM	>	Dimension of DMATX.
integer	MGDIM	>	Dimension of AMATX.
integer	NLARGE	>	Large deformation flag. Large deformation analysis if NLARGE=1 and infinitesimal deformation analysis otherwise.
integer	NTYPE	>	Stress state type flag.
double precision	AMATX	<	Consistent spatial tangent modulus.
double precision	DMATX	<	Consistent infinitesimal tangent modulus.
double precision	EINCR	>	Current incremental engineering strain components. Used only in small strain analysis.
double precision	FINCR	>	Current incremental deformation gradient. Used only in large strain analysis.
integer	IPROPS	>	Array of integer material properties.
logical	LALGVA	>	Array of current logical algorithmic variables at the Gauss point of interest.
double precision	RALGVA	>	Array of current real algorithmic variables at the Gauss points of interest.
double precision	RPROPS	>	Array of real material properties.
double precision	RSTAVA	>	Array of current real state variables at the Gauss points of interest.
double precision	RSTAV2	>	Array of real state variables at the Gauss point of interest at the previous equilibrium (converged) configuration.
double precision	STRES	>	Array of current (Cauchy) stress components at the current Gauss point.

SUBOUTINE MATIOR - MANUAL PAGE

DESCRIPTION
Material interface for output result routine calls

This routine calls the material-specific routines to output results according to the material type.

ARGUMENT LIST

Type	Name		Description
integer	NTYPE	>	Stress state type flag.
integer	IPROPS	>	Array of integer material properties.
double precision	RALGVA	>	Array of current real algorithmic variables.
double precision	RPROPS	>	Array of real material properties.
double precision	RSTAVA	>	Array of current real state variables.
double precision	STRES	>	Array of current (Cauchy) stress components.

SUBOUTINE MATIRD - MANUAL PAGE

DESCRIPTION
Material interface for reading material-specific input data

This routine calls the routines that read material-specific input data from the data file.

ARGUMENT LIST

Type	Name	Description
character	MATNAM >	Character string containing the material name.
integer	NLARGE >	Large strain analysis flag.
integer	NTYPE >	Stress state type flag.
logical	UNSAUX <	Logical unsymetric stiffness flag.
integer	IPROPS <	Array of integer material properties.
double precision	RPROPS <	Array of real material properties.

SUBOUTINE MATISU - MANUAL PAGE

DESCRIPTION
Material interface for state update routine calls

This routine calls the state update routine according to the material type. Given the material-independent kinematic quantities computed at the element level (and passed into the present routine through its list of arguments) this routine identifies the material type in question and calls the corresponding material-specific state update routine which updates the stress and other state variables at one Gauss point.

ARGUMENT LIST

Type	Name		Description
double precision	DETF	>	Determinant of the total deformation gradient at the current Gauss point. Used for large strain analysis only.
integer	NLARGE	>	Large strain flag. Large strain analysis if NLARGE=1 and infinitesimal strain analysis otherwise.
integer	NTYPE	>	Stress state type flag.
logical	SUFAIL	<	State update failure flag. Return value set to .FALSE. if the state update procedure was performed successfully and set to .TRUE. otherwise.
double precision	THKGP	<>	Thickness of the current Gauss point. Used only in plane stress analysis. Updated here only in large strain analysis. In this case, it is the initial (reference config.) thickness on entry and returns as the current thickness.
double precision	EINCR	>	Array of incremental engineering strain components. Used in infinitesimal strain analysis only.
double precision	FINCR	>	Incremental deformation gradient. Used in large strain analysis only.
integer	IPROPS	>	Array of integer material properties.
logical	LALGVA	<>	Array of current logical algorithmic variables at the current Gauss point.
double precision	RALGVA	<>	Array of current real algorithmic variables at the current Gauss point. Previous converged (equilibrium) value on entry. Returns as updated value.
double precision	RPROPS	>	Array of real material properties.
double precision	RSTAVA	<>	Array of current real state variables at the current Gauss point. Previous converged (equilibrium) value on entry. Returns as updated value.
double precision	STRES	<>	Array of current (Cauchy) stress components at the current Gauss point. Previous converged (equilibrium) value on entry. Returns as updated value.

SUBOUTINE MATISW - MANUAL PAGE

DESCRIPTION
Material interface for initialisation and switching state variables

This routine calls the state/algorithmic variables initialising/switching routines according to the material type. Each material type has its own routine that initialises and switches Gauss point data (between current and last converged values).
The initialised/switched data comprises state and algorithmic variables whose initialisation/switching rules depend on the particular material type considered.

ARGUMENT LIST

Type	Name		Description
integer	MODE	>	Initialisation/switching mode flag.
integer	NLARGE	>	Large strain flag. Large strain analysis if NLARGE=1 and infinitesimal strain analysis otherwise.
integer	NTYPE	>	Stress state type flag.
integer	IPROPS	>	Array of integer material properties.
logical	LALGVC	<>	Array of current logical algorithmic variables at the current Gauss point.
logical	LALGVL	<>	Array of logical algorithmic variables at the last equilibrium configuration for the Gauss point in question.
double precision	RALGVC	<>	Array of current real algorithmic variables at the current Gauss point.
double precision	RALGVL	<>	Array of real algorithmic variables at the last equilibrium configuration for the current Gauss point.
double precision	RPROPS	>	Array of real material properties.
double precision	RSTAVC	<>	Array of current real state variables at the current Gauss point.
double precision	RSTAVL	<>	Array of real state variables at the last equilibrium configuration at the current Gauss point.
double precision	STRESC	<>	Array of current (Cauchy) stress components at the current Gauss point.
double precision	STRESL	<>	Array of (Cauchy) stress components at last equilibrium configuration for the current Gauss point.

INTEGER FUNCTION NFUNC - MANUAL PAGE

DESCRIPTION
Simple integer calculation used in frontal slover

SUBOUTINE NODAVE - MANUAL PAGE

DESCRIPTION
Computes and prints averaged nodal values of Gauss-point variables.

This routine computes and prints the averaged (smoothed) nodal values of stresses and other state and algorithmic variables. The smoothed value of each variable at a node is computed by firstly extrapolating the value of the variable from the Gauss-points to the node in question and then averaging the extrapolated values obtained from all elements sharing that node. The values computed here are used in HYPLAS only for the purpose of output.

INTEGER FUNCTION NWORD - MANUAL PAGE

DESCRIPTION
Returns the number of words contained in a character string

The return value of this function is the number of words contained in the character string passed in its argument list. The function also sets the pointers to the beginning and end of each word. The pointers are returned via argument list.

ARGUMENT LIST

Type	Name		Description
character	CHRSTR	>	Character string.
integer	IWBEG	<	Array of pointers to the beginning of the words contained in CHRSTR. For the Nth word, beginning at CHRSTR(I:I), the function sets IWBEG(N)=I.
integer	IWEND	<	Array of pointers to the end of the words contained in CHRSTR. For the Nth word, ending at CHRSTR(I:I), the function sets IWEND(N)=I.

SUBOUTINE ORDAMA - MANUAL PAGE

DESCRIPTION
Output results Lemaitre's ductile damage elasto-plastic model

This routine writes to the results file the internal and algorithmic variables of Lemaitre's ductile damage elasto-plastic material with non-linear (piece-wise linear) isotropic hardening.

ARGUMENT LIST

Type	Name		Description
double precision	DGAMA	>	Incremental plastic multiplier. Computed in routine SUDAMA.
integer	NOUTF	>	Results file unit identifier.
integer	NTYPE	>	Stress state type flag.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components.
double precision	STRES	>	Array of stress tensor components.

SUBOUTINE ORDMEL - MANUAL PAGE

DESCRIPTION
Output results for the damaged elastic/crack closure model.

This routine writes to the results file some state variables for the isotropically damaged isotropic elastic model accounting for partial microcrack/void closure effects (quasi-unilateral conditions) under compressive stresses.

ARGUMENT LIST

Type	Name		Description
integer	NOUTF	>	Results file unit identifier.
integer	NTYPE	>	Stress state type flag.
double precision	STRES	>	Array of stress tensor components.

SUBOUTINE ORDP - MANUAL PAGE

DESCRIPTION
Output results for the Drucker-Prager elasto-plastic model

This routine writes to the results file the internal and
algorithmic variables of the Drucker-Prager elasto-plastic
material model with non-linear isotropic hardening.
The results printed here are obtained in routine SUCADP.

ARGUMENT LIST

Type	Name		Description
double precision	DGAM	>	Array of incremental plastic multipliers.
integer	NOUTF	>	Results file unit identifier.
integer	NTYPE	>	Stress state type flag.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components.
double precision	STRES	>	Array of stress tensor components.

SUBOUTINE OREL - MANUAL PAGE

DESCRIPTION
Output results for the linear elastic material model

This routine writes to the results file some state variables for the linear elastic material model.

ARGUMENT LIST

Type	Name		Description
integer	NOUTF	>	Results file unit identifier.
integer	NTYPE	>	Stress state type flag.
double precision	STRES	>	Array of stress tensor components.

SUBOUTINE ORMC - MANUAL PAGE

DESCRIPTION
Output results for the Mohr-Coulomb elasto-plastic material model

This routine writes to the results file the internal and algorithmic variables of the Mohr-Coulomb elasto-plastic material with non-linear isotropic hardening. The results printed here are obtained by the return mapping algorithm implemented in routine SUMC.

ARGUMENT LIST

Type	Name		Description
double precision	DGAM	>	Array of incremental plastic multipliers. Computed in routine SUMC.
integer	NOUTF	>	Results file unit identifier.
integer	NTYPE	>	Stress state type flag.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components.
double precision	STRES	>	Array of stress tensor components.

SUBOUTINE OROGD - MANUAL PAGE

DESCRIPTION
Output results for the Ogden hyperelastic material model

This routine writes to the results file state variables for the Ogden hyperelastic material model. The state update procedure for this material model is carried out in subroutine SUOGD.

ARGUMENT LIST

Type	Name		Description
integer	NOUTF	>	Results file unit identifier.
integer	NTYPE	>	Stress state type flag.
double precision	STRES	>	Array of stress tensor components.

SUBOUTINE ORPDSC - MANUAL PAGE

DESCRIPTION
Output results for the planar double-slip single crystal model

This routine writes to the results file the internal and algorithmic variables of the planar double-slip single crystal elasto-plastic material model with non-linear isotropic Taylor hardening.

ARGUMENT LIST

Type	Name		Description
double precision	DGAM	>	Array of incremental plastic multipliers. Computed in routine SUTR.
integer	NOUTF	>	Results file unit identifier.
integer	NTYPE	>	Stress state type flag.
double precision	RPROPS	>	Array of real material properties.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components.
double precision	STRES	>	Array of stress tensor components.

SUBOUTINE ORTR - MANUAL PAGE

DESCRIPTION
Output results for the Tresca elasto-plastic material model

This routine writes to the results file the internal and algorithmic variables of the Tresca elasto-plastic material model with non-linear isotropic hardening. The results printed here are the von Mises effective stress, the equivalent plastic strain and the incremental plastic multiplier(s) obtained by the return mapping algorithm of routine SUTR.

ARGUMENT LIST

Type	Name		Description
double precision	DGAM	>	Array of incremental plastic multipliers. Computed in routine SUTR.
integer	NOUTF	>	Results file unit identifier.
integer	NTYPE	>	Stress state type flag.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components.
double precision	STRES	>	Array of stress tensor components.

SUBOUTINE ORVM - MANUAL PAGE

DESCRIPTION
Output results for the von Mises elasto-plastic material model

This routine writes to the results file the internal and algorithmic variables of the von Mises elasto-plastic material with non-linear isotropic hardening. The results printed here are the von Mises effective stress, the equivalent plastic strain and the incremental plastic multiplier obtained by the return mapping algorithm of routine SUVM or SUVMPS.

ARGUMENT LIST

Type	Name		Description
double precision	DGAMA	>	Incremental plastic multiplier. Computed in routine SUVM or SUVMPS.
integer	NOUTF	>	Results file unit identifier.
integer	NTYPE	>	Stress state type flag.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components.
double precision	STRES	>	Array of stress tensor components.

SUBOUTINE ORVMMX - MANUAL PAGE

DESCRIPTION
Output results for the von Mises model with mixed hardening

This routine writes to the results file the internal and algorithmic variables of the von Mises elasto-plastic material with non-linear mixed hardening.

ARGUMENT LIST

Type	Name		Description
double precision	DGAMA	>	Incremental plastic multiplier. Computed in routine SUVMMX.
integer	NOUTF	>	Results file unit identifier.
integer	NTYPE	>	Stress state type flag.
double precision	RSTAVA	>	Array of real state variables other than the stress tensor components.
double precision	STRES	>	Array of stress tensor components.

SUBOUTINE OUTPUT - MANUAL PAGE

DESCRIPTION
Prints displacements, reactions and other variables to results file

This routine prints the nodal displacements and reactions and Gauss-point and nodal averaged values of stresses and other state and algorithmic variables to the results file. The argument NOUTP controls what results are to be printed out.

ARGUMENT LIST

Type	Name		Description
double precision	TFACT	>	Total proportional load factor.
integer	IINCS	>	Current load increment number.
integer	IITER	>	Equilibrium iteration at which convergence was achieved in the current load step.
integer	NOUTP	>	Load output flag that determines what results are to be printed out.

SUBOUTINE PEXIT - MANUAL PAGE

DESCRIPTION
Aborts the execution of HYPLAS

This routine closes the open files and stops the execution of HYPLAS, sending a message to the results file and to the standard output. It is called in emergency situations when a irrecoverable error occurs.

DOUBLE PRECISION FUNCTION PLFUN - MANUAL PAGE

DESCRIPTION
Returns the value of a piece-wise linear scalar function

This function returns the value of a piece-wise linear scalar function F(X) defined by a set of NPOINT pairs (X,F(X)) passed in the matrix argument XFX.

ARGUMENT LIST

Type	Name		Description
double precision	X	>	Point at which the function will be evaluated.
integer	NPOINT	>	Number of points defining the piece-wise linear function.
double precision	XFX	>	Matrix (dimension 2*NPOINT) containing the pairs (x,f(x)) which define the piece-wise linear function. evaluated. Each column of XFX contains a pair (xi,f(xi)). The pairs supplied in XFX must be ordered such that the x's are monotonically increasing. That is, the x [XFX(1,i+1)] of a column i+1 must be greater than XFX(1,i) (x of column i). If X < XFX(1,1) the piece-wise linear function is assumed constant equal to XFX(1,1). If X > XFX(1,NPOINT) the piece-wise linear function is assumed constant equal to XFX(1,NPOINT).

SUBOUTINE PODEC2 - MANUAL PAGE

DESCRIPTION
Polar decomposition of 2-D tensors

This routine performs the right polar decomposition of 2-D tensors: F = R U, where R is a rotation (orthogonal tensor) and U is a symmetric tensor.

ARGUMENT LIST

Type	Name		Description
double precision	F	>	2-D tensor to be decomposed. Dimension 2x2.
double precision	R	<	Rotation matrix resulting from the polar decomposition. Dimension 2x2.
double precision	U	<	Right symmetric tensor resulting from the polar decomposition. Dimension 2x2.

SUBOUTINE PRINC2 - MANUAL PAGE

DESCRIPTION
Computes the principal stresses and their angle in 2-D.

This routine computes the eigenvalues of the stress tensor in 2-D
and the angle of the system of eingenvectors relative to the
system where the stress components are expressed.

ARGUMENT LIST

Type	Name		Description
double precision	PSTRS	<	Array of principal stresses and angle.
double precision	STRSG	>	Array of stress components.

SUBOUTINE RDDAMA - MANUAL PAGE

DESCRIPTION

Reads data for Lemaitre's ductile damage elasto-plastic model.

This routine reads from the data file and echos to the results file the material parameters necessary for Lemaitre's ductile damage elasto-plastic model with piece-wise linear isotropic hardening.
It also sets the array of real properties and some components of the array of integer material properties.
These arrays are used by subroutines SUDAMA and CTDAMA.
This routine also checks whether the number of integer and real material properties, the number of real state variables and the number of logical algorithmic variables required by the present model is compatible with the dimension of the corresponding global arrays of HYPLAS.
It also sets the unsymmetric tangent stiffness flag.

ARGUMENT LIST

Type	Name	Description
integer	IPROPS <	Array of integer material properties.
integer	MIPROP >	Dimension of the global array of integer material properties.
integer	MLALGV >	Dimension of the global array of logical algorithmic variables.
integer	MRPROP >	Dimension of the global array of real material variables.
integer	MRSTAV >	Dimension of the global array of real state variables.
integer	NTYPE >	Stress state type.
double precision	RPROPS <	Array of real material properties.
logical	UNSYM <	Unsymmetric tangent stiffness flag.

SUBOUTINE RDDMEL - MANUAL PAGE

DESCRIPTION
Read data for damaged elastic/crack closure model.

This routine reads from the data file and echos to the results file the material parameters necessary for the isotropically damaged isotropic elastic model accounting for partial microcrack/void closure effects (quasi-unilateral conditions) under compressive stresses.

ARGUMENT LIST

Type	Name		Description
integer	NTYPE	>	Stress state type flag.
double precision	RPROPS	<	Array of real material properties.
logical	UNSYM	<	Unsymmetric tangent stiffness flag.

SUBOUTINE RDDP - MANUAL PAGE

DESCRIPTION
Read data for the Drucker-Prager material model.

This routine reads from the data file and echos to the results file the material parameters necessary for the Drucker-Prager elasto-plastic model with piece-wise linear isotropic hardening.
It also sets the array of real properties and some components of the array of integer material properties. These arrays are used by subroutines SUDP and CTDP.
It also sets the unsymmetric tangent stiffness flag.

ARGUMENT LIST

Type	Name	Description
integer	IPROPS <	Array of integer material properties.
integer	MIPROP >	Dimension of the global array of integer material properties.
integer	MLALGV >	Dimension of the global array of logical algorithmic variables.
double precision	MRALGV >	Dimension of the global array of real algorithmic variables.
integer	MRPROP >	Dimension of the global array of real material variables.
integer	MRSTAV >	Dimension of the global array of real state variables.
double precision	RPROPS <	Array of real material properties.
logical	UNSYM <	Unsymmetric tangent stiffness flag.

SUBOUTINE RDEL - MANUAL PAGE

DESCRIPTION
Read data for the linear elastic material model.

This routine reads from the data file and echos to the results file the material parameters necessary for the linear elastic material model.

ARGUMENT LIST

Type	Name	Description
integer	MRPROP >	Dimension of the global array of real material variables.
integer	MRSTAV >	Dimension of the global array of real state variables.
double precision	RPROPS <	Array of real material properties.
logical	UNSYM <	Unsymmetric tangent stiffness flag.

SUBOUTINE RDMC - MANUAL PAGE

DESCRIPTION
Read data for the Mohr-Coulomb elasto-plastic material model.

This routine reads from the data file and echos to the results file the material parameters necessary for the Mohr-Coulomb elasto-plastic model with piece-wise linear isotropic hardening. It also sets the array of real properties and some components of the array of integer material properties. These arrays are used by subroutines SUMC and CTMC. It also sets the unsymmetric tangent stiffness flag.

ARGUMENT LIST

Type	Name	Description
integer	IPROPS <	Array of integer material properties.
integer	MIPROP >	Dimension of the global array of integer material properties.
integer	MLALGV >	Dimension of the global array of logical algorithmic variables.
double precision	MRALGV >	Dimension of the global array of real algorithmic variables.
integer	MRPROP >	Dimension of the global array of real material variables.
integer	MRSTAV >	Dimension of the global array of real state variables.
double precision	RPROPS <	Array of real material properties.
logical	UNSYM <	Unsymmetric tangent stiffness flag.

SUBOUTINE RDOGD - MANUAL PAGE

DESCRIPTION
Reads data for the regularised Ogden hyperelastic material model.

This routine reads from the data file and echoes to the results file the material parameters required by the regularised Ogden hyperelastic material model. It also sets the array of real properties and some components of the array of integer material properties. These arrays are used by subroutines SUOGD and CTOGD. Note that under plane stress the model implemented here is exactly incompressible, whereas under plane strain and axisymmetric conditions the regularised (penalty type) approach is used to enforce incompressibility. This routine also sets the unsymmetric tangent stiffness flag.

ARGUMENT LIST

Type	Name	Description
integer	IPROPS <	Array of integer material properties.
integer	MIPROP >	Dimension of the global array of integer material properties.
integer	MRPROP >	Dimension of the global array of real material variables.
integer	MRSTAV >	Dimension of the global array of real state variables.
double precision	RPROPS <	Array of real material properties.
logical	UNSYM <	Unsymmetric tangent stiffness flag. Always returned as .FALSE..

SUBOUTINE RDPDSC - MANUAL PAGE

DESCRIPTION
Read data for planar double-slip single crystal model.

This routine reads from the data file and echos to the results file
the material parameters necessary for the large strain
planar double-slip single crystal elasto-plastic model with
piece-wise linear isotropic Taylor hardening.
It also sets the array of real properties and some components of
the array of integer material properties.
These arrays are used by subroutines SUPDSC and CSTPDS.
This routine also checks whether the number of integer and real
material properties, the number of real state variables and the
number of logical and real algorithmic variables required by the
present model is compatible with the dimension of the corresponding
global arrays.
It also sets the unsymmetric tangent stiffness flag.

ARGUMENT LIST

Type	Name	Description
integer	IPROPS <	Array of integer material properties.
integer	MIPROP >	Dimension of the global array of integer material properties.
integer	MLALGV >	Dimension of the global array of logical algorithmic variables.
integer	MRALGV >	Dimension of the global array of real algorithmic variables.
integer	MRPROP >	Dimension of the global array of real material variables.
integer	MRSTAV >	Dimension of the global array of real state variables.
integer	NLARGE >	Large strain analysis integer flag. Used for checking only. Present model is compatible only with large strain analysis.
integer	NTYPE >	Stress state type integer flag. Used for checking only. Present model is compatible only with plane strain analysis (NTYPE=2).
double precision	RPROPS <	Array of real material properties.
logical	UNSYM <	Unsymmetric tangent stiffness flag.

SUBOUTINE RDTR - MANUAL PAGE

DESCRIPTION
Reads data for the Tresca elasto-plastic material model.

This routine reads from the data file and echos to the results file the material parameters necessary for the Tresca elasto-plastic model with piece-wise linear isotropic hardening.
It also sets the array of real properties and some components of the array of integer material properties.
These arrays are used by subroutines SUTR and CTTR.
This routine also checks whether the number of integer and real material properties, the number of real state variables and the number of logical algorithmic variables required by the present model is compatible with the dimension of the corresponding global arrays.
It also sets the unsymmetric tangent stiffness flag.

ARGUMENT LIST

Type	Name	Description
integer	IPROPS <	Array of integer material properties.
integer	MIPROP >	Dimension of the global array of integer material properties.
integer	MLALGV >	Dimension of the global array of logical algorithmic variables.
integer	MRPROP >	Dimension of the global array of real material variables.
integer	MRSTAV >	Dimension of the global array of real state variables.
integer	NLARGE >	Large deformation analysis flag.
integer	NTYPE >	Stress state type flag.
double precision	RPROPS <	Array of real material properties.
logical	UNSYM <	Unsymmetric tangent stiffness flag.

SUBOUTINE RDVM - MANUAL PAGE

DESCRIPTION
Read data for von Mises elasto-plastic material model.

This routine reads from the data file and echos to the results file the material parameters necessary for the von Mises elasto-plastic model with piece-wise linear isotropic hardening.
It also sets the array of real properties and some components of the array of integer material properties.
These arrays are used by subroutines SUVM or SUVMPS and CTVM or CTVMPS.
This routine also checks whether the number of integer and real material properties, the number of real state variables and the number of logical algorithmic variables required by the present model
is compatible with the dimension of the corresponding global arrays.
It also sets the unsymmetric tangent stiffness flag.

ARGUMENT LIST

Type	Name	Description
integer	IPROPS <	Array of integer material properties.
integer	MIPROP >	Dimension of the global array of integer material properties.
integer	MLALGV >	Dimension of the global array of logical algorithmic variables.
integer	MRPROP >	Dimension of the global array of real material variables.
integer	MRSTAV >	Dimension of the global array of real state variables.
double precision	RPROPS <	Array of real material properties.
logical	UNSYM <	Unsymmetric tangent stiffness flag.

SUBOUTINE RDVMMX - MANUAL PAGE

DESCRIPTION
Read data for the von Mises plasticity model with mixed hardening.

This routine reads from the data file and echos to the results file
the material parameters necessary for
the von Mises elasto-plastic model with piece-wise linear
mixed isotropic/kinematic hardening.
It also sets the array of real properties and some components of
the array of integer material properties.
These arrays are used by subroutines SUVMMX and CTVMMX.
This routine also checks whether the number of integer and real
material properties, the number of real state variables and the
number of logical algorithmic variables required by the present
model
is compatible with the dimension of the corresponding global
arrays.
It also sets the unsymmetric tangent stiffness flag.

ARGUMENT LIST

Type	Name	Description
integer	IPROPS <	Array of integer material properties.
integer	MIPROP >	Dimension of the global array of integer material properties.
integer	MLALGV >	Dimension of the global array of logical algorithmic variables.
integer	MRPROP >	Dimension of the global array of real material variables.
integer	MRSTAV >	Dimension of the global array of real state variables.
integer	NLARGE >	Large strain analysis flag.
integer	NTYPE >	Stress state type flag.
double precision	RPROPS <	Array of real material properties.
logical	UNSYM <	Unsymmetric tangent stiffness flag.

SUBOUTINE RSQ4 - MANUAL PAGE

DESCRIPTION
Read input and set properties for element type QUAD4

This routine reads data from the input data file and sets the element properties arrays for elements type QUAD4: Standard isoparametric 4-noded quadrilateral for plane strain, plane stress and axisymmetric analysis. It also echoes the properties to the results file and sets the unsymmetric tangent stiffness flag.

ARGUMENT LIST

Type	Name		Description
integer	IELPRP	<	Array of integer element properties.
integer	NDATF	>	Data file unit identifier.
integer	NRESF	>	Results file unit identifier.
double precision	RELPRP	<	Array of real element properties.
logical	UNSYM	<	Unsymmetric tangent stiffness flag.

SUBOUTINE RSQ4FB - MANUAL PAGE

DESCRIPTION
Read input and set properties for element type QUA4FB

This routine reads data from the input data file and sets the element properties arrays for elements type QUA4FB: F-bar 4-noded quadrilateral for plane strain, plane stress and axisymmetric analysis. It also echoes the properties to the results file and sets the unsymmetric tangent stiffness flag.

ARGUMENT LIST

Type	Name		Description
integer	IELPRP	<	Array of integer element properties.
integer	NDATF	>	Data file unit identifier.
integer	NRESF	>	Results file unit identifier.
integer	NTYPE	>	Stress state type.
double precision	RELPRP	<	Array of real element properties.
logical	UNSYM	<	Unsymmetric tangent stiffness flag.

SUBOUTINE RSQ8 - MANUAL PAGE

DESCRIPTION
Read input and set properties for element type QUAD8

This routine reads data from the input data file and sets the element properties arrays for elements type QUAD8: Standard isoparametric 8-noded quadrilateral for plane strain, plane stress and axisymmetric analysis. It also echoes the properties to the results file and sets the unsymmetric tangent stiffness flag.

ARGUMENT LIST

Type	Name		Description
integer	IELPRP	<	Array of integer element properties.
integer	NDATF	>	Data file unit identifier.
integer	NRESF	>	Results file unit identifier.
double precision	RELPRP	<	Array of real element properties.
logical	UNSYM	<	Unsymmetric tangent stiffness flag.

SUBOUTINE RST3 - MANUAL PAGE

DESCRIPTION
Set properties arrays for element type TRI3

This routine sets the element properties arrays for elements type TRI3: Standard isoparametric 3-noded triangle for plane strain, plane stress and axisymmetric analysis. It also echoes the properties to the results file and sets the unsymmetric tangent stiffness flag.

ARGUMENT LIST

Type	Name		Description
integer	IELPRP	<	Array of integer element properties.
integer	NRESF	>	Results file unit identifier.
double precision	RELPRP	<	Array of real element properties.
logical	UNSYM	<	Unsymmetric tangent stiffness flag.

SUBOUTINE RSTART - MANUAL PAGE

DESCRIPTION
Reads (writes) data to input (output) re-start file

Depending on the entry value of its argument MODE, this routine either dumps the complete database and current information about the analysis into an output re-start file or reads this information from a previously created re-start file.
The information dumped on the output re-start file will enable HYPLAS to re-start from the point in the analysis where the file was generated. To re-start, this file will be read as the input re-start file.

ARGUMENT LIST

Type	Name		Description
double precision	DFOLD	<>	Incremental load factor of the previous load increment.
double precision	DLENG	<>	Current arc-length (used for arc-length method only).
double precision	DLENGO	<>	Previous arc-length (used for arc-length method only).
double precision	DLENM	<>	Current maximum arc-length (used for arc-length method only).
double precision	DLAMD	<>	Iterative load factor obtained by the arc-length method only.
integer	IFNEG	<>	Signum (+1/-1) of the tangent stifness matrix determinant (used only by the arc-length method).
integer	IINCS	<>	Number of the current global (equilibrium iteration).
integer	MXFRON	<>	Maximum front encoutered.
integer	NOUTP	>	Array of output flags for the current load increment.
double precision	TFACT	<>	Current total load factor.
double precision	TFACTO	<>	Previous total load factor.
logical	UNSYM	<>	Global unsymmetric solver flag.
character	RSTINP	>	Current input re-start file name.
character	RSTOUT	>	Current output re-start file name.
integer	MODE	>	If MODE=0, reads from current input re-start file. If MODE=1, writes to the current output re-start file.
integer	INCRST	<	Increment number corresponding to the current output re-start file.

SUBOUTINE RSTCHK - MANUAL PAGE

DESCRIPTION
Checks wether the main data is to be read from a re-start file

This routine reads the data file and checks wether the main data is to be read from it or from a re-start file.

ARGUMENT LIST

Type	Name	Description
character	RSTINP >	Character string containing the input re-start file name.
logical	RSTRT <	Restart flag. Return value is set to .FALSE. if the main data is to be read from the data file. Set to .TRUE. if the main data is to be read from a re-start file.

SUBOUTINE RTSR - MANUAL PAGE

DESCRIPTION
Matrix product s.Rt S R

This routine performs the matrix product s Rt S R, where s is a scalar, R a rectangular real matrix and S a square real matrix. Rt denotes the transpose of R.

ARGUMENT LIST

Type	Name		Description
double precision	AUXM	<	Auxiliary matrix used to store partial results of the calculation.
integer	MODE	>	If set to 1, the argument Q returns the resulting matrix Rt S R. Otherwise, Rt S R is added to the input value of Q.
integer	MROWQ	>	Dimensioning parameter: maximum dimension of the square matrix Q.
integer	MROWR	>	Dimensioning parameter: maximum number of rows of R (same as the maximum dimension of square matrix S).
integer	NCOLR	>	Number of columns of R.
integer	NROWR	>	Number of rows of R.
double precision	Q	<>	Matrix where results are stored.
double precision	R	>	Rectangular real matrix.
double precision	S	>	Square real matrix.
double precision	SCAL	>	Real scalar.
logical	UNSYM	>	Unsymmetry flag.

SUBOUTINE RTSX - MANUAL PAGE

DESCRIPTION
Matrix product s.Rt S X

This routine performs the matrix product s Rt S X, where s is a scalar, R and X rectangular real matrices of identical dimensions and S a square real matrix. Rt denotes the transpose of R.

ARGUMENT LIST

Type	Name		Description
double precision	AUXM	<	Auxiliary matrix used to store partial results of the calculation.
integer	MODE	>	If set to 1, the argument Q returns the resulting matrix Rt S R. Otherwise, Rt S R is added to the input value of Q.
integer	MROWQ	>	Dimensioning parameter: maximum dimension of the square matrix Q.
integer	MROWR	>	Dimensioning parameter: maximum number of rows of R (same as the maximum dimension of square matrix S).
integer	NCOLR	>	Number of columns of R.
integer	NROWR	>	Number of rows of R.
double precision	Q	<>	Matrix where results are stored.
double precision	R	>	Rectangular real matrix.
double precision	S	>	Square real matrix.
double precision	X	>	Rectangular real matrix.
double precision	SCAL	>	Real scalar.

SUBOUTINE RTV - MANUAL PAGE

DESCRIPTION
Matrix-vectot product s.Rt V

This routine performs the matrix-vector product s Rt V, where s is a scalar, R a real rectangular matrix and v a real vector. Rt denotes the transpose of R.

ARGUMENT LIST

Type	Name		Description
integer	MODE	>	If set to 1, the argument V returns the resulting vector s Rt V. Otherwise, s Rt v is added to the input value of P.
integer	MROWR	>	Dimensioning parameter: maximum number of rows of R.
integer	NCOLR	>	Number of columns of R.
integer	NROWR	>	Number of rows of R.
double precision	P	<>	Vector where results are stored.
double precision	R	>	Rectangular real matrix.
double precision	V	>	Real vector.
double precision	SCAL	>	Real scalar.

SUBOUTINE RVSCAL - MANUAL PAGE

DESCRIPTION
Multiplies a double precision vector by a scalar.

ARGUMENT LIST

Type	Name		Description
double precision	V	<>	Double precision vector.
integer	N	>	Dimension of V.
double precision	SCAL	>	Scalar by which V will be multiplied.

SUBOUTINE RVSUB - MANUAL PAGE

DESCRIPTION
Subtracts two double precision vectors

This function subtracts two double precision vectors passed as arguments and stores the result in another vector U=V-W.

ARGUMENT LIST

Type	Name		Description
double precision	U	<	Double precision vector with the result V-W.
double precision	V	>	Double precision vector.
double precision	W	>	Double precision vector.

SUBOUTINE RVZERO - MANUAL PAGE

DESCRIPTION
Zero a double precision array

This routine sets to zero the N components of the double precision array argument V.

ARGUMENT LIST

Type	Name		Description
double precision	V	<	Zeroed double precision array.
integer	N	>	Dimension of V.

DOUBLE PRECISION FUNCTION SCAPRD - MANUAL PAGE

DESCRIPTION
Scalar product of double precision vectors

This function returns the scalar product between its two double precision vector arguments U and V.

ARGUMENT LIST

Type	Name		Description
double precision	U	>	Array of components of a double precision vector.
double precision	V	>	Array of components of a double precision vector.
integer	N	>	Dimension of U and V.

SUBOUTINE SETBE - MANUAL PAGE

DESCRIPTION
Obtains the left Cauchy-Green strain tensor from the log strain

Given the Eulerian (Lagrangian) logarithmic strain tensor, this routine computes the corresponding left (right) Cauchy-Green strain tensor.
Plane strain, plane stress and axisymmetric implementations.
In the present implementation, the out-of-plane component is always computed.

ARGUMENT LIST

Type	Name		Description
double precision	BE	<>	Array of engineering logarithmic strain components on entry. Returns as the array of components of the corresponding Cauchy-Green strain tensor.
integer	NTYPE	>	Stress state type flag.

SUBOUTINE SFQ4 - MANUAL PAGE

DESCRIPTION
Shape function and derivatives for element type QUAD4

This routine computes the shape functions and shape function derivatives for the element type QUAD4: Standard isoparametric 4-noded quadrilateral for plane strain, plane stress and axisymmetric analysis.

ARGUMENT LIST

Type	Name		Description
double precision	DERIV	<	Array of shape function derivatives.
double precision	ETASP	>	Isoparametric coordinate ETA of the point where the shape functions or their derivatives are to be evaluated.
double precision	EXISP	>	Isoparametric coordinate XI of the point where the shape functions or their derivatives are to be evaluated.
integer	IBOUND	>	Boundary interpolation flag. Entry must be 0 for domain interpolation. Boundary interpolation is assumed otherwise.
integer	MDIME	>	Maximum permissible number of spatial dimensions. Used here only for array dimensioning.
double precision	SHAPE	<	Array of shape function values.

SUBOUTINE SFQ4FB - MANUAL PAGE

DESCRIPTION
Shape function and derivatives for element type QUA4FB

This routine computes the shape functions and shape function derivatives for the element type QUA4FB: F-bar isoparametric 4-noded quadrilateral for plane strain and axisymmetric analysis.

ARGUMENT LIST

Type	Name		Description
double precision	DERIV	<	Array of shape function derivatives.
double precision	ETASP	>	Isoparametric coordinate ETA of the point where the shape functions or their derivatives are to be evaluated.
double precision	EXISP	>	Isoparametric coordinate XI of the point where the shape functions or their derivatives are to be evaluated.
integer	IBOUND	>	Boundary interpolation flag. Entry must be 0 for domain interpolation. Boundary interpolation is assumed otherwise.
integer	MDIME	>	Maximum permissible number of spatial dimensions. Used here only for array dimensioning.
double precision	SHAPE	<	Array of shape function values.

SUBOUTINE SFQ8 - MANUAL PAGE

DESCRIPTION
Shape function and derivatives for element type QUAD8

This routine computes the shape functions and shape function derivatives for the element type QUAD8: Standard isoparametric 8-noded quadrilateral for plane strain, plane stress and axisymmetric analysis.

ARGUMENT LIST

Type	Name		Description
double precision	DERIV	<	Array of shape function derivatives.
double precision	ETASP	>	Isoparametric coordinate ETA of the point where the shape functions or their derivatives are to be evaluated.
double precision	EXISP	>	Isoparametric coordinate XI of the point where the shape functions or their derivatives are to be evaluated.
integer	IBOUND	>	Boundary interpolation flag. Entry must be 0 for domain interpolation. Boundary interpolation is assumed otherwise.
integer	MDIME	>	Maximum permissible number of spatial dimensions. Used here only for array dimensioning.
double precision	SHAPE	<	Array of shape function values.

SUBOUTINE SFT3 - MANUAL PAGE

DESCRIPTION
Shape function and derivatives for element type TRI3

This routine computes the shape functions and shape function derivatives for the element type TRI3: Standard isoparametric 3-noded triangle for plane strain, plane stress and axisymmetric analysis.

ARGUMENT LIST

Type	Name		Description
double precision	DERIV	<	Array of shape function derivatives.
double precision	ETASP	>	Isoparametric coordinate ETA of the point where the shape functions or their derivatives are to be evaluated.
double precision	EXISP	>	Isoparametric coordinate XI of the point where the shape functions or their derivatives are to be evaluated.
integer	IBOUND	>	Boundary interpolation flag. Entry must be 0 for domain interpolation. Boundary interpolation is assumed otherwise.
integer	MDIME	>	Maximum permissible number of spatial dimensions. Used here only for array dimensioning.
double precision	SHAPE	<	Array of shape function values.

SUBOUTINE SHPFUN - MANUAL PAGE

DESCRIPTION
Call shape function/derivative computation routines

This routine calls the shape function/shape function derivative computation routines for all element types. This routine needs to be modified for inclusion of new elements in HYPLAS.

ARGUMENT LIST

Type	Name		Description
double precision	DERIV	<	Array of shape function derivatives.
double precision	ETASP	>	Isoparametric coordinate ETA of the point where the shape functions or their derivatives are to be evaluated.
double precision	EXISP	>	Isoparametric coordinate XI of the point where the shape functions or their derivatives are to be evaluated.
integer	IBOUND	>	Boundary interpolation flag. Entry must be 0 for domain interpolation. Boundary interpolation is assumed otherwise.
integer	IELTYP	>	Element type flag. Follows enumeration in file ELEMENTS.INC.
integer	MDIME	>	Maximum permissible number of spatial dimensions. Used here only for array dimensioning.
double precision	SHAPE	<	Array of shape function values.

SUBOUTINE SOLQUA - MANUAL PAGE

DESCRIPTION
Solves a quadratic equation: a x**2 + b x + c = 0

Given the coeficients a, b and c, this routine computes the real roots of the associated quadratic equation, a x**2 + b x + c = 0.
The return values of the arguments ROOT1 and ROOT2 (the roots) are set only if real roots exist.
If the equation admits only one real solution, the return value of the logical argument ONEROO is set to .TRUE. (set to .FALSE. otherwise).
If the equation admits two real roots, the return value of the logical argument TWOROO is set to .TRUE. (set to .FALSE. otherwise).
Consequently, if the roots are complex or no roots/infinite number of roots exist (the equation is ill-defined), the return value of the logical arguments ONEROO and TWOROO is set to .FALSE.

ARGUMENT LIST

Type	Name		Description
double precision	A	>	Coefficient of the quadratic term.
double precision	B	>	Coefficient of the linear term.
double precision	C	>	Coefficient of the constant term.
logical	ONEROO	<	Logical flag. Set to .TRUE. if there is only one (real) root. Set to .FALSE. otherwise.
logical	TWOROO	<	Logical flag. Set to .TRUE. if there are two distinct (real) root. Set to .FALSE. otherwise.
double precision	ROOT1	<	One of the roots (the only root if there is only one real root - not set if there are no real roots).
double precision	ROOT2	<	The other root (not set if there is only one real root or no real roots).

SUBOUTINE SPDEC2 - MANUAL PAGE

DESCRIPTION
Closed form spectral decomposition of 2-D symmetric tensors

This routine performs the spectral decomposition of 2-D symmetric tensors in closed form. The tensor is passed as argument (stored in vector form).

ARGUMENT LIST

Type	Name		Description
double precision	EIGPRJ	<	Matrix with one eigenprojection tensor of X stored in each column.
double precision	EIGX	<	Array containing the eigenvalues of X.
logical	REPEAT	<	Repeated eigenvalues flag. Set to .TRUE. if the eigenvalues of X are repeated (within a small tolerance).
double precision	X	>	Array containing the components of a symmetric tensor.

SUBOUTINE STFBA2 - MANUAL PAGE

DESCRIPTION
Stiffness matrix computation for elements of class FBAR

This routine computes the tangent stiffness matrix for all elements of the class FBAR: F-bar 2-D elements for plane strain and axisymmetric analysis.

ARGUMENT LIST

Type	Name		Description
integer	IELEM	>	Number of the element whose stiffness matrix is to be computed.
integer	KUNLD	>	Unloading flag. KUNLD is set to 1 if the loading programme is currently unloading.
integer	MDIME	>	Maximum permissible number of spatial dimensions.
integer	MELEM	>	Maximum permissible number of elements.
integer	MPOIN	>	Maximum permissible number of nodal points in a mesh.
integer	MSTRE	>	Maximum permissible number of stress (or resultant force) components.
integer	MTOTV	>	Maximum permissible number degrees of freedom in a mesh.
integer	NAXIS	>	Symmetry axis flag. Used only for axisymmetric analysis. NAXIS=1 if symmetric about the X-axis and NAXIS=2 if symmetric about the Y-axis.
integer	NTYPE	>	Stress state type flag.
logical	UNSYM	>	Tangent stiffness symmetry flag. .TRUE. if tangent stiffness is unsymmetric, .FALSE. otherwise.
double precision	COORD1	>	Array of coordinates of all nodes of the mesh in the current configuration.
double precision	DINCR	>	Global array of incremental displacements of the entire mesh.
double precision	ESTIF	<	Tangent stiffness matrix of the current element.
integer	IELPRP	>	Array of integer element properties of the current element group.
integer	IPROPS	>	Array of integer material properties of the current element group.
logical	LALGVA	>	Array of current logical algorithmic variables of all Gauss points of the current element.
integer	LNODS	>	Global array of connectivity of the entire mesh.
double precision	RALGVA	>	Array of current real algorithmic variables of all Gauss points of the current element.
double precision	RELPRP	>	Array of real element properties of the current element group.
double precision	RPROPS	>	Array of real material properties of the current element group.
double precision	RSTAVA	>	Array of current real state variables of all Gauss points of the current element.
double precision	RSTAV2	>	Array of real state variables of all Gauss points of the current element at the previous equilibrium (converged) configuration.

double precision	STRSG	>	Array of current stress components of all Gauss points of the current element.
double precision	TDISP	>	Global array of total displacement of all nodes of the mesh.

SUBOUTINE STSTD2 - MANUAL PAGE

DESCRIPTION
Stiffness matrix computation for elements of class STDARD

This routine computes the tangent stiffness matrix for all elements of the class STDARD: standard isoparametric 2-D elements for plane strain, plane stress and axisymmetric analysis.

ARGUMENT LIST

Type	Name		Description
integer	IELEM	>	Number of the element whose stiffness matrix is to be computed.
integer	KUNLD	>	Unloading flag. KUNLD is set to 1 if the loading programme is currently unloading.
integer	MDIME	>	Maximum permissible number of spatial dimensions.
integer	MELEM	>	Maximum permissible number of elements.
integer	MPOIN	>	Maximum permissible number of nodal points in a mesh.
integer	MSTRE	>	Maximum permissible number of stress (or resultant force) components.
integer	MTOTV	>	Maximum permissible number degrees of freedom in a mesh.
integer	NAXIS	>	Symmetry axis flag. Used only for axisymmetric analysis. NAXIS=1 if symmetric about the X-axis and NAXIS=2 if symmetric about the Y-axis.
integer	NLARGE	>	Large deformation flag. Large deformation analysis if NLARGE=1 and infinitesimal deformation analysis otherwise.
integer	NTYPE	>	Stress state type flag.
logical	UNSYM	>	Tangent stiffness symmetry flag. .TRUE. if tangent stiffness is unsymmetric, .FALSE. otherwise.
double precision	COORD1	>	Array of coordinates of all nodes of the mesh in the current configuration.
double precision	DINCR	>	Global array of incremental displacements of the entire mesh.
double precision	ESTIF	<	Tangent stiffness matrix of the current element.
integer	IELPRP	>	Array of integer element properties of the current element group.
integer	IPROPS	>	Array of integer material properties of the current element group.
logical	LALGVA	>	Array of current logical algorithmic variables of all Gauss points of the current element.
integer	LNODS	>	Global array of connectivity of the entire mesh.
double precision	RALGVA	>	Array of current real algorithmic variables of all Gauss points of the current element.
double precision	RELPRP	>	Array of real element properties of the current element group.
double precision	RPROPS	>	Array of real material properties of the current element group.
double precision	RSTAVA	>	Array of current real state variables of all Gauss points of the current element.

```
double precision   RSTAV2 >  Array of real state
                             variables of all Gauss points of the
                             current element at the previous
                             equilibrium (converged) configuration.

double precision   STRSG  >  Array of current stress components of
                             all Gauss points of the current element.

double precision   THKGP  >  Array of current thickness of
                             all Gauss points of the current element.
                             Used only in plane stress analysis.

double precision   TDISP  >  Global array of total displacement
                             of all nodes of the mesh.
```

SUBOUTINE SUDAMA - MANUAL PAGE

DESCRIPTION
State update procedure for Lemaitre's ductile damage model.

This routine uses the fully implicit elastic predictor/return mapping algorithm as the state update procedure for Lemaitre's ductile damage elasto-plastic model with general non-linear (piece-wise linear) isotropic hardening under plane strain and axisymmetric conditions.

ARGUMENT LIST

Type	Name		Description
double precision	DGAMA	<	Incremental plastic multiplier.
integer	IPROPS	>	Array of integer material properties. The number of points on the piece-wise linear hardening curve is the only element of this array used here. This array is set in routines INDATA and RDDAMA.
logical	LALGVA	<	Array of logical algorithmic flags. For the present material model, this array contains the plastic yielding flag and the return algorithm failure flag. The plastic yielding flag is set to .TRUE. if plastic yielding has occurred and to .FALSE. if the step is elastic. The algorithm failure flag is set to .FALSE. if the state update algorithm has been successful and to .TRUE. if the return mapping algorithm has failed to converge.
integer	NTYPE	>	Stress state type flag. The present routine is compatible only with NTYPE=2 (plane strain) and NTYPE=3 (axisymmetric condition).
double precision	RPROPS	>	Array of real material properties. This array contains the density (not used in this routine), the elastic properties: Young's modulus and Poisson's ratio, the two damage evolution constants and the pairs ``hardening variable strain-uniaxial yield stress'' defining the (user supplied) piece-wise linear hardening curve. This array is set in routine RDDAMA.
double precision	RSTAVA	<>	Array of real state variables other than the stress tensor components. Previous converged values on entry, updated values on exit. The state variables stored in this array are the (engineering) elastic strain components, the hardening variable and the damage internal variable.
double precision	STRAT	>	Array of elastic trial (engineering) strain components.
double precision	STRES	<	Array of updated stress tensor components.

SUBOUTINE SUDMEL - MANUAL PAGE

DESCRIPTION
State update procedure for damaged elastic/crack closure model.

Given the total strain, this routine computes the corresponding stress using for an isotropically damaged isotropic linear elastic model accounting for partial microcrack/void closure effects (quasi-unilateral conditions) under compressive stresses.
This routine contains the plane strain and axisymmetric implementations of the model.

ARGUMENT LIST

Type	Name		Description
integer	NTYPE	>	Stress state type flag. The present routine is compatible with NTYPE=1 (plane stress), NTYPE=2 (plane strain) and NTYPE=3 (axisymmetric condition).
double precision	RPROPS	>	Array of real material properties.
double precision	RSTAVA	<	Array of real state variables other than the stress tensor components. For this model , this array stores the (engineering) strain components.
double precision	STRAN	>	Array of current total (engineering) strain components.
double precision	STRES	<	Array of updated stress tensor components.
logical	SUFAIL	<	Logical flag for stress update procedure failure. Set to .TRUE. if the algorithm failed to update the stress. Set to .FALSE. is the stress was successfully updated.

SUBOUTINE SUDP - MANUAL PAGE

DESCRIPTION

State update procedure for the Drucker-Prager material model.

This routine uses the fully implicit elastic predictor/return mapping algorithm as the state update procedure for the Drucker elasto-plastic material model with piece-wise linear isotropic hardening. This routine contains only the plane strain and axisymmetric implementations of the model.

ARGUMENT LIST

Type	Name		Description
double precision	DGAM	<	Incremental plastic multiplier.
integer	IPROPS	>	Array of integer material properties. The number of points on the piece-wise linear hardening curve is the only element stored in this array used here. This array is set in routines INDATA and RDDP.
logical	LALGVA	<	Array of logical algorithmic flags. For the present material model, this array contains the plastic yielding flag, IFPLAS; the return algorithm failure flag, SUFAIL; the apex return flag, APEX. The plastic yielding flag is set to .TRUE. if plastic yielding has occurred and to .FALSE. if the step is elastic. The algorithm failure flag is set to .FALSE. if the state update algorithm has been successful and to .TRUE. if the return mapping algorithm has failed to converge. APEX is set to .TRUE. if the selected return mapping is the return to the apex. It is set to .FALSE. otherwise.
integer	NTYPE	>	Stress state type flag.
double precision	RPROPS	>	Array of real material properties. This array contains the density (not used in this routine), the elastic properties: Young's modulus and Poisson's ratio, and the plastic properties: ETA, XI, ETABAR and the pairs ``accumulated plastic strain-cohesion'' defining the (user supplied) piece-wise linear hardening curve. This array is set in routine RDDP.
double precision	RSTAVA	<>	Array of real state variables other than the stress tensor components. Previous converged values on entry, updated values on exit. The state variables stored in this array are the (engineering) elastic strain components and the accumulated plastic strain.
double precision	STRAT	>	Array of elastic trial (engineering) strain components.
double precision	STRES	<	Array of updated stress tensor components.

SUBOUTINE SUDPPN - MANUAL PAGE

DESCRIPTION
State update procedure for the Drucker-Prager model. Plane stress.

This routine uses the fully implicit elastic predictor/return mapping algorithm as the state update procedure for the Drucker-Prager elasto-plastic material model with general non-linear (piece-wise linear) isotropic hardening under plane stress condition.
The algorithm used here is based on the nested iteration approach for enforcement of the plane stress constraint at the Gauss point level.

ARGUMENT LIST

Type	Name	Description
double precision	RALGVA <	Array of real algorithmic variables. For the present plane stress implementation, it contains the incremental plastic multipliers and the elastic trial thickness strain obtained as the solution of the plane stress enforcement loop.
integer	IPROPS >	Array of integer material properties. The number of points on the piece-wise linear hardening curve is the only element stored in this array used here. This array is set in routines INDATA and RDDP.
logical	LALGVA <	Array of logical algorithmic flags. For the present material model, this array contains the plastic yielding flag, IFPLAS; the return algorithm failure flag, SUFAIL; the apex return flag, APEX. The plastic yielding flag is set to .TRUE. if plastic yielding has occurred and to .FALSE. if the step is elastic. The algorithm failure flag is set to .FALSE. if the state update algorithm has been successful and to .TRUE. if the return mapping algorithm has failed to converge. APEX is set to .TRUE. if the selected return mapping is the return to the apex. It is set to .FALSE. otherwise.
integer	NTYPE >	Stress state type flag. The present routine is compatible only with NTYPE=1 (plane stress).
double precision	RPROPS >	Array of real material properties. This array contains the density (not used in this routine), the elastic properties: Young's modulus and Poisson's ratio, and the plastic properties: ETA, XI, ETABAR and the pairs ``accumulated plastic strain-cohesion'' defining the (user supplied) piece-wise linear hardening curve. This array is set in routine RDDP.
double precision	RSTAVA <>	Array of real state variables other than the stress tensor components. Previous converged values on entry, updated values on exit. The state variables stored in this array are the (engineering) elastic strain components and the accumulated plastic strain.
double precision	STRAT <>	Array of elastic trial (engineering) strain components. Its first three components are the in-plane elastic trial components which are not updated in the present routine. Its fourth component - the thickness elastic trial strain - is determined here as the solution to the plane stress enforcement N-R loop.

double precision	STRES	<	Array of updated stress tensor components.

SUBOUTINE SUEL - MANUAL PAGE

DESCRIPTION
State update procedure for the linear elastic material model.

Given the total strain, this routine computes the corresponding stress using the standard generalised Hooke's law for linear elastic materials. This routine contains the plane stress, plane strain and axisymmetric implementations of the linear elastic model.

ARGUMENT LIST

Type	Name		Description
integer	NTYPE	>	Stress state type flag. The present routine is compatible with NTYPE=1 (plane stress), NTYPE=2 (plane strain) and NTYPE=3 (axisymmetric condition).
double precision	RPROPS	>	Array of real material properties.
double precision	RSTAVA	<	Array of real state variables other than the stress tensor components. For the linear elastic model, this array stores the (engineering) strain components.
double precision	STRAN	>	Array of current total (engineering) strain components.
double precision	STRES	<	Array of updated stress tensor components.

SUBOUTINE SUMC - MANUAL PAGE

DESCRIPTION
State update procedure for the Mohr-Coulomb type material model.

This routine uses the fully implicit elastic predictor/return mapping algorithm as the state update procedure for the Mohr-Coulomb elasto-plastic material model with piece-wise linear isotropic hardening. It contains the plane strain and axisymmetric implementations of the model. The essential return mapping is carried out here in principal stress space.

ARGUMENT LIST

Type	Name		Description
double precision	DGAM	<	Array of incremental plastic multipliers.
integer	IPROPS	>	Array of integer material properties. The number of points on the piece-wise linear hardening curve is the only element of this array used here. This array is set in routines INDATA and RDMC.
logical	LALGVA	<	Array of logical algorithmic flags. For the present material model, this array contains the plastic yielding flag, IFPLAS; the return algorithm failure flag, SUFAIL; the edge return flag, EDGE; the right edge return flag, RIGHT, and; the apex return flag, APEX. The plastic yielding flag is set to .TRUE. if plastic yielding has occurred and to .FALSE. if the step is elastic. The algorithm failure flag is set to .FALSE. if the state update algorithm has been successful and to .TRUE. if the return mapping algorithm has failed to converge. EDGE is set to .TRUE. if the selected return mapping is to and edge (right or left) and is set to .FALSE. otherwise. RIGHT is set to .TRUE. if the selected return is to the right edge and is set to .FALSE. otherwise. APEX is set to .TRUE. if the selected return is to the apex and is .FALSE. otherwise.
integer	NTYPE	>	Stress state type flag. This routine is compatible only with plane strain (NTYPE=2) and axisymmetric states (NTYPE=3).
double precision	RPROPS	>	Array of real material properties. This array contains the density (not used in this routine), the elastic properties: Young's modulus and Poisson's ratio, and the plastic properties: Sine and cosine of frictional angle, cosine of the dilatancy angle and the pairs ``accumulated plastic strain-cohesion'' defining the (user supplied) piece-wise linear hardening curve. This array is set in routine RDMC.
double precision	RSTAVA	<>	Array of real state variables other than the stress tensor components. Previous converged values on entry, updated values on exit. The state variables stored in this array are the (engineering) elastic strain components and the accumulated plastic strain.
double precision	STRAT	>	Array of elastic trial (engineering) strain components.
double precision	STRES	<	Array of updated stress tensor components.

SUBOUTINE SUOGD - MANUAL PAGE

DESCRIPTION
State update procedure for the Ogden hyperelastic material model.

This routine updates the Cauchy stress and other state variables for the regularised (compressible) Ogden hyperelastic material model under plane stress, plane strain and axisymmetric states. Under plane stress, the model is exactly incompressible rather than regularised. Under plane strain and axisymmetric conditions, the regularised (penalty type) approach is adopted to enforce incompressibility. In the regularised approach, the bulk modulus plays the role of the incompressibility penalty factor.

ARGUMENT LIST

Type	Name		Description
double precision	B	>	Array of components of the Left Cauchy-Green strain tensor.
integer	IPROPS	>	Array of integer material properties.
integer	NTYPE	>	Stress state type flag. This routine accepts NTYPE=1 (plane stress) NTYPE=2 (plane strain) and NTYPE=3 (axisymmetric).
double precision	RPROPS	>	Array of real material properties.
double precision	RSTAVA	<	Array of real state variables other than stress tensor components. Use here to store the left Cauchy-Green strain tensor.
double precision	STRES	<	Array of Cauchy stress components.
double precision	THICK	<>	Thickness of the current point. Initial (reference) value on entry, current updated value on return. Used only in plane stress analysis.

SUBOUTINE SUPDSC - MANUAL PAGE

DESCRIPTION
State update for the planar double-slip single crystal model.

This routine uses an implicit elastic predictor/return mapping algorithm as the state update procedure for the anisotropic finite strain planar double-slip single crystal model with Taylor isotropic hardening. This model is compatible only with the plane strain assumption.
It has two slip planes comprising a total of four slip directions (two directions opposite to each other in each plane).

ARGUMENT LIST

Type	Name		Description
double precision	DGAM	<	Array of incremental plastic multipliers.
double precision	FINCR	>	Incremental deformation gradient.
integer	IPROPS	>	Array of integer material properties. The number of points on the piece-wise linear hardening curve is the only element of this array used here. This array is set in routines INDATA and RDTR.
logical	LALGVA	<	Array of logical algorithmic flags. For the present material model, this array contains the plastic yielding flag, IFPLAS, the state update failure flag, SUFAIL and the active slip-system flags, S1ACT, S2ACT, S3ACT and S4ACT. The plastic yielding flag is set to .TRUE. if plastic yielding has occurred and to .FALSE. if the step is elastic. The algorithm failure flag is set to .FALSE. if the state update algorithm has been successful and to .TRUE. if the return mapping algorithm has failed for any reason. Each active slip-system flags is set to .TRUE. only if the corresponding system is currently active, i.e., S1ACT is set to .TRUE. if system 1 is active, S2ACT is set to .TRUE. if system 2 is active and so on. Each flag will be set to .FALSE. if the corresponding system is not currently active.
integer	NTYPE	>	Stress state type flag. The present implementation is compatible only with plane strain (NTYPE=2).
double precision	RPROPS	>	Array of real material properties. This array contains the density (not used in this routine), the elastic properties: Shear and bulk modulus (neo-Hookean constants) and the plastic properties: the pairs ``Taylor hardening variable-resolved yield shear stress'' that define the (user supplied) piece-wise linear Taylor hardening curve. This array is set in routine RDPDSC.
double precision	RSTAVA	<>	Array of real state variables other than the stress tensor components. Previous converged values on entry, updated values on exit. The state variables stored in this array are the in-plane components of the elastic deformation gradient and the Taylor hardening internal variable.
double precision	STRES	<	Array of updated stress tensor components.

SUBOUTINE SUTR - MANUAL PAGE

DESCRIPTION

State update procedure for the Tresca elasto-plastic model.

This routine uses the fully implicit elastic predictor/return mapping algorithm as the state update procedure for the Tresca elasto-plastic material model with piece-wise linear isotropic hardening. It contains the plane strain and the axisymmetric implementations of the model.
The essential return mapping is carried out here in the principal stress space.

ARGUMENT LIST

Type	Name		Description
double precision	DGAM	<	Array of incremental plastic multipliers.
integer	IPROPS	>	Array of integer material properties. The number of points on the piece-wise linear hardening curve is the only element of this array used here. This array is set in routines INDATA and RDTR.
logical	LALGVA	<	Array of logical algorithmic flags. For the present material model, this array contains the plastic yielding flag, IFPLAS; the return algorithm failure flag, SUFAIL; the two-vector return flag, TWOVEC and the right corner return flag, RIGHT. The plastic yielding flag is set to .TRUE. if plastic yielding has occurred and to .FALSE. if the step is elastic. The algorithm failure flag is set to .FALSE. if the state update algorithm has been successful and to .TRUE. if the return mapping algorithm has failed to converge. TWOVEC is set to .TRUE. if the selected return mapping is to a corner (right or left) and is set to .FALSE. otherwise. RIGHT is set to .TRUE. if the selected return is to the right corner and is set to .FALSE. otherwise.
integer	NTYPE	>	Stress state type flag.
double precision	RPROPS	>	Array of real material properties. This array contains the density (not used in this routine), the elastic properties: Young's modulus and Poisson's ratio, and the plastic properties: the pairs ``accumulated plastic strain-cohesion'' defining the (user supplied) piece-wise linear hardening curve. This array is set in routine RDTR.
double precision	RSTAVA	<>	Array of real state variables other than the stress tensor components. Previous converged values on entry, updated values on exit. The state variables stored in this array are the (engineering) elastic strain components and the accumulated plastic strain.
double precision	STRAT	>	Array of elastic trial (engineering) strain components.
double precision	STRES	<	Array of updated stress tensor components.

SUBOUTINE SUTRPN - MANUAL PAGE

DESCRIPTION
State update procedure for the Tresca model in plane stress.

This routine uses the fully implicit elastic predictor/return mapping algorithm as the state update procedure for the Tresca elasto-plastic material model with piece-wise linear isotropic hardening under plane stress condition.
The algorithm used here is based on the nested iteration approach for enforcement of the plane stress constraint at the Gauss point level.

ARGUMENT LIST

Type	Name		Description
double precision	DGAM	<	Array of incremental plastic multipliers.
integer	IPROPS	>	Array of integer material properties. The number of points on the piece-wise linear hardening curve is the only element of this array used here. This array is set in routines INDATA and RDTR.
logical	LALGVA	<	Array of logical algorithmic flags. For the present material model, this array contains the plastic yielding flag, IFPLAS; the return algorithm failure flag, SUFAIL; the two-vector return flag, TWOVEC and the right corner return flag, RIGHT. The plastic yielding flag is set to .TRUE. if plastic yielding has occurred and to .FALSE. if the step is elastic. The algorithm failure flag is set to .FALSE. if the state update algorithm has been successful and to .TRUE. if the return mapping algorithm has failed to converge. TWOVEC is set to .TRUE. if the selected return mapping is to a corner (right or left) and is set to .FALSE. otherwise. RIGHT is set to .TRUE. if the selected return is to the right corner and is set to .FALSE. otherwise.
integer	NTYPE	>	Stress state type flag.
double precision	RPROPS	>	Array of real material properties. This array contains the density (not used in this routine), the elastic properties: Young's modulus and Poisson's ratio, and the plastic properties: the pairs ``accumulated plastic strain-cohesion'' defining the (user supplied) piece-wise linear hardening curve. This array is set in routine RDTR.
double precision	RSTAVA	<>	Array of real state variables other than the stress tensor components. Previous converged values on entry, updated values on exit. The state variables stored in this array are the (engineering) elastic strain components and the accumulated plastic strain.
double precision	STRAT	>	Array of elastic trial (engineering) strain components.
double precision	STRES	<	Array of updated stress tensor components.

SUBOUTINE SUVM - MANUAL PAGE

DESCRIPTION
State update procedure for the von Mises material model.

This routine uses the fully implicit elastic predictor/return mapping algorithm as the state update procedure for the von Mises elasto-plastic material model with general non-linear (piece-wise linear) isotropic hardening under plane strain or axisymmetric conditions.

ARGUMENT LIST

Type	Name		Description
double precision	DGAMA	<	Incremental plastic multiplier.
integer	IPROPS	>	Array of integer material properties. The number of points on the piece-wise linear hardening curve is the only element of this array used here. This array is set in routines INDATA and RDVM.
logical	LALGVA	<	Array of logical algorithmic flags. For the present material model, this array contains the plastic yielding flag and the return algorithm failure flag. The plastic yielding flag is set to .TRUE. if plastic yielding has occurred and to .FALSE. if the step is elastic. The algorithm failure flag is set to .FALSE. if the state update algorithm has been successful and to .TRUE. if the return mapping algorithm has failed to converge.
integer	NTYPE	>	Stress state type flag. The present routine is compatible only with NTYPE=2 (plane strain) and NTYPE=3 (axisymmetric condition).
double precision	RPROPS	>	Array of real material properties. This array contains the density (not used in this routine), the elastic properties: Young's modulus and Poisson's ratio, and the pairs ``accumulated plastic strain-uniaxial yield stress'' defining the (user supplied) piece-wise linear hardening curve. This array is set in routine RDVM.
double precision	RSTAVA	<>	Array of real state variables other than the stress tensor components. Previous converged values on entry, updated values on exit. The state variables stored in this array are the (engineering) elastic strain components and the accumulated plastic strain.
double precision	STRAT	>	Array of elastic trial (engineering) strain components.
double precision	STRES	<	Array of updated stress tensor components.

SUBOUTINE SUVMMX - MANUAL PAGE

DESCRIPTION

State update for the von Mises model with mixed hardening.

This routine uses the fully implicit elastic predictor/return mapping algorithm as the state update procedure for the von Mises elasto-plastic material model with general non-linear (piece-wise linear) mixed isotropic/kinematic hardening under plane strain and axisymmetric conditions.

ARGUMENT LIST

Type	Name		Description
double precision	DGAMA	<	Incremental plastic multiplier.
integer	IPROPS	>	Array of integer material properties. The number of points on the piece-wise linear hardening curves is the only element of this array used here. This array is set in routines MATIRD and RDVMMX.
logical	LALGVA	<	Array of logical algorithmic flags. For the present material model, this array contains the plastic yielding flag and the return algorithm failure flag. The plastic yielding flag is set to .TRUE. if plastic yielding has occurred and to .FALSE. if the step is elastic. The algorithm failure flag is set to .FALSE. if the state update algorithm has been successful and to .TRUE. if the return mapping algorithm has failed to converge.
integer	NTYPE	>	Stress state type flag. The present routine is compatible only with NTYPE=2 (plane strain) and NTYPE=3 (axisymmetric condition).
double precision	RPROPS	>	Array of real material properties. This array contains the density (not used in this routine), the elastic properties: Young's modulus and Poisson's ratio, the pairs ``accumulated plastic strain-isotropic hard. stress'' defining the isotropic hardening curve and the pairs ``accumulated plastic strain-kinematic hardening stress'' defining the kinematic hardening curve. This array is set in routine RDVMMX.
double precision	RSTAVA	<>	Array of real state variables other than the stress tensor components. Previous converged values on entry, updated values on exit. The state variables stored in this array are the (engineering) elastic strain components, the accumulated plastic strain and the backstress tensor components.
double precision	STRAT	>	Array of elastic trial (engineering) strain components.
double precision	STRES	<	Array of updated stress tensor components.

SUBOUTINE SUVMPS - MANUAL PAGE

DESCRIPTION
State update procedure for the von Mises model in plane stress.

This routine uses the fully implicit elastic predictor/return mapping algorithm as the state update procedure for the von Mises elasto-plastic material model with general non-linear (piece-wise linear) isotropic hardening under plane stress condition.
The algorithm used here is derived from the plane stress-projected von Mises model, in which only in-plane stress/strain components take part in the evolution equations.

ARGUMENT LIST

Type	Name		Description
double precision	DGAMA	<	Incremental plastic multiplier.
integer	IPROPS	>	Array of integer material properties. The number of points on the piece-wise linear hardening curve is the only element of this array used here. This array is set in routines INDATA and RDVM.
logical	LALGVA	<	Array of logical algorithmic flags. For the present material model, this array contains the plastic yielding flag and the return algorithm failure flag. The plastic yielding flag is set to .TRUE. if plastic yielding has occurred and to .FALSE. if the step is elastic. The algorithm failure flag is set to .FALSE. if the state update algorithm has been successful and to .TRUE. if the return mapping algorithm has failed to converge.
integer	NTYPE	>	Stress state type flag. The present routine is compatible only with NTYPE=1 (plane stress).
double precision	RPROPS	>	Array of real material properties. This array contains the density (not used in this routine), the elastic properties: Young's modulus and Poisson's ratio, and the pairs ``accumulated plastic strain-uniaxial yield stress'' defining the (user supplied) piece-wise linear hardening curve. This array is set in routine RDVM.
double precision	RSTAVA	<>	Array of real state variables other than the stress tensor components. Previous converged values on entry, updated values on exit. The state variables stored in this array are the (engineering) elastic strain components and the accumulated plastic strain.
double precision	STRAT	>	Array of in-plane elastic trial (engineering) strain components.
double precision	STRES	<	Array of updated stress tensor components.

SUBOUTINE SWDAMA - MANUAL PAGE

DESCRIPTION
Initialise/switch state variables for Lemaitre's damage model.

This routine initialises and switches state variables (between current and previous values) for Lemaitre's ductile damage elasto-plastic material model with isotropic hardening.

ARGUMENT LIST

Type	Name		Description
integer	MODE	>	Initialisation/Switching mode.
integer	NTYPE	>	Stress state type flag.
double precision	LALGVC	<>	Array of logical algorithmic variables at Gauss point. Current values.
double precision	LALGVL	<>	Array of logical algorithmic variables at Gauss point. Last converged (equilibrium) values.
double precision	RALGVC	<>	Array of real algorithmic variables at Gauss point. Current values.
double precision	RSTAVC	<>	Array of real state variables at Gauss point. Current values.
double precision	RSTAVL	<>	Array of real state variables at Gauss point. Last converged (equilibrium) values.
double precision	STRESC	<>	Array of stress (Cauchy in large strain) components. Current values.
double precision	STRESL	<>	Array of stress (Cauchy in large strain) components. Last converged (equilibrium) values.

SUBOUTINE SWDMEL - MANUAL PAGE

DESCRIPTION
Initialise/switch state variables for damaged elastic model

This initialises and switches state variables (between current and previous values) for the damaged elastic material model (damaged Hencky material in large strain analysis) with microcrack/void closure effects.

ARGUMENT LIST

Type	Name		Description
integer	MODE	>	Initialisation/Switching mode.
integer	NTYPE	>	Stress state type flag.
double precision	RSTAVC	<>	Array of real state variables at Gauss point. Current values.
double precision	RSTAVL	<>	Array of real state variables at Gauss point. Last converged (equilibrium) values.
double precision	STRESC	<>	Array of stress (Cauchy in large strain) components. Current values.
double precision	STRESL	<>	Array of stress (Cauchy in large strain) components. Last converged (equilibrium) values.

SUBOUTINE SWDP - MANUAL PAGE

DESCRIPTION
Initialise/switch state variables for the Drucker-Prager material

This initialises and switches state variables (between current and previous values) for the Drucker-Prager elasto-plastic material model.

ARGUMENT LIST

Type	Name		Description
integer	MODE	>	Initialisation/Switching mode.
integer	NTYPE	>	Stress state type flag.
double precision	LALGVC	<>	Array of logical algorithmic variables at Gauss point. Current values.
double precision	LALGVL	<>	Array of logical algorithmic variables at Gauss point. Last converged (equilibrium) values.
double precision	RALGVC	<>	Array of real algorithmic variables at Gauss point. Current values.
double precision	RSTAVC	<>	Array of real state variables at Gauss point. Current values.
double precision	RSTAVL	<>	Array of real state variables at Gauss point. Last converged (equilibrium) values.
double precision	STRESC	<>	Array of stress (Cauchy in large strain) components. Current values.
double precision	STRESL	<>	Array of stress (Cauchy in large strain) components. Last converged (equilibrium) values.

SUBOUTINE SWEL - MANUAL PAGE

DESCRIPTION
Initialise/switch state variables for the elastic material model

This initialises and switches state variables (between current and previous values) for the elastic material model (Hencky material in large strain analysis).

ARGUMENT LIST

Type	Name		Description
integer	MODE	>	Initialisation/Switching mode.
integer	NTYPE	>	Stress state type flag.
double precision	RSTAVC	<>	Array of real state variables at Gauss point. Current values.
double precision	RSTAVL	<>	Array of real state variables at Gauss point. Last converged (equilibrium) values.
double precision	STRESC	<>	Array of stress (Cauchy in large strain) components. Current values.
double precision	STRESL	<>	Array of stress (Cauchy in large strain) components. Last converged (equilibrium) values.

SUBOUTINE SWITCH - MANUAL PAGE

DESCRIPTION
Switches the stored value of the problem variables.

This routine switches the stored values of the problem variables (displacements, coordinates, stresses and other state and algorithmic variables) between current values and values at the last converged (equilibrium) solution. The value of the argument MODE defines what switching operation is carried out. If MODE=1, the current value of the variables is assigned to the converged solution and the equilibrium values of the previous step are discarded (this is required when convergence is achieved at the end of the current equilibrium iteration). If MODE=2, the values at the last converged solution are assigned to the current values and the values at the end of the previous iteration (if any) are discarded (this operation is carried out whenever a new iteration is required by the iterative method for equilibrium solution). If MODE=3, the values at the last converged solution are assigned to the current values when increment cutting is required. Gauss-point thicknesses are also switched accordingly (for large strains under plane stress only).

ARGUMENT LIST

Type	Name		Description
integer	MODE	>	Flag determining which switching operation is to be carried out.

SUBOUTINE SWMC - MANUAL PAGE

DESCRIPTION
Initialise/switch state variables for the Mohr-Coulomb material

This initialises and switches state variables (between current and previous values) for the Mohr-Coulomb elasto-plastic material model.

ARGUMENT LIST

Type	Name		Description
integer	MODE	>	Initialisation/Switching mode.
integer	NTYPE	>	Stress state type flag.
double precision	LALGVC	<>	Array of logical algorithmic variables at Gauss point. Current values.
double precision	LALGVL	<>	Array of logical algorithmic variables at Gauss point. Last converged (equilibrium) values.
double precision	RALGVC	<>	Array of real algorithmic variables at Gauss point. Current values.
double precision	RSTAVC	<>	Array of real state variables at Gauss point. Current values.
double precision	RSTAVL	<>	Array of real state variables at Gauss point. Last converged (equilibrium) values.
double precision	STRESC	<>	Array of stress (Cauchy in large strain) components. Current values.
double precision	STRESL	<>	Array of stress (Cauchy in large strain) components. Last converged (equilibrium) values.

SUBOUTINE SWOGD - MANUAL PAGE

DESCRIPTION
Initialise/switch state variables for the Ogden hyperelastic model

This initialises and switches state variables (between current and previous values) for the Ogden hyperelastic material model.

ARGUMENT LIST

Type	Name		Description
integer	MODE	>	Initialisation/Switching mode. See source code comments for details.
integer	NTYPE	>	Stress state type flag.
double precision	RSTAVC	<>	Array of real state variables at Gauss point. Current values.
double precision	RSTAVL	<>	Array of real state variables at Gauss point. Last converged (equilibrium) values.
double precision	STRESC	<>	Array of stress (Cauchy in large strain) components. Current values.
double precision	STRESL	<>	Array of stress (Cauchy in large strain) components. Last converged (equilibrium) values.

SUBOUTINE SWPDSC - MANUAL PAGE

DESCRIPTION
Initialise/switch state variables for the planar single crystal

This initialises and switches state variables (between current and previous values) for the planar double-slip single crystal elasto-plastic model.

ARGUMENT LIST

Type	Name		Description
integer	MODE	>	Initialisation/Switching mode.
double precision	LALGVC	<>	Array of logical algorithmic variables at Gauss point. Current values.
double precision	LALGVL	<>	Array of logical algorithmic variables at Gauss point. Last converged (equilibrium) values.
double precision	RALGVC	<>	Array of real algorithmic variables at Gauss point. Current values.
double precision	RSTAVC	<>	Array of real state variables at Gauss point. Current values.
double precision	RSTAVL	<>	Array of real state variables at Gauss point. Last converged (equilibrium) values.
double precision	STRESC	<>	Array of stress (Cauchy in large strain) components. Current values.
double precision	STRESL	<>	Array of stress (Cauchy in large strain) components. Last converged (equilibrium) values.

SUBOUTINE SWTR - MANUAL PAGE

DESCRIPTION
Initialise/switch state variables for Tresca plasticity model

This initialises and switches state variables (between current and previous values) for the Tresca elasto-plastic material model.

ARGUMENT LIST

Type	Name		Description
integer	MODE	>	Initialisation/Switching mode.
integer	NTYPE	>	Stress state type flag.
double precision	LALGVC	<>	Array of logical algorithmic variables at Gauss point. Current values.
double precision	LALGVL	<>	Array of logical algorithmic variables at Gauss point. Last converged (equilibrium) values.
double precision	RALGVC	<>	Array of real algorithmic variables at Gauss point. Current values.
double precision	RSTAVC	<>	Array of real state variables at Gauss point. Current values.
double precision	RSTAVL	<>	Array of real state variables at Gauss point. Last converged (equilibrium) values.
double precision	STRESC	<>	Array of stress (Cauchy in large strain) components. Current values.
double precision	STRESL	<>	Array of stress (Cauchy in large strain) components. Last converged (equilibrium) values.

SUBOUTINE SWVM - MANUAL PAGE

DESCRIPTION
Initialise/switch state variables for the von Mises model

This initialises and switches state variables (between current and previous values) for the von Mises elasto-plastic material model with isotropic hardening.

ARGUMENT LIST

Type	Name		Description
integer	MODE	>	Initialisation/Switching mode.
integer	NTYPE	>	Stress state type flag.
double precision	LALGVC	<>	Array of logical algorithmic variables at Gauss point. Current values.
double precision	LALGVL	<>	Array of logical algorithmic variables at Gauss point. Last converged (equilibrium) values.
double precision	RALGVC	<>	Array of real algorithmic variables at Gauss point. Current values.
double precision	RSTAVC	<>	Array of real state variables at Gauss point. Current values.
double precision	RSTAVL	<>	Array of real state variables at Gauss point. Last converged (equilibrium) values.
double precision	STRESC	<>	Array of stress (Cauchy in large strain) components. Current values.
double precision	STRESL	<>	Array of stress (Cauchy in large strain) components. Last converged (equilibrium) values.

SUBOUTINE SWVMMX - MANUAL PAGE

DESCRIPTION
Initialise/switch variables for the mixed hardening von Mises model

This initialises and switches state variables (between current and previous values) for the von Mises elasto-plastic material model with mixed isotropic/kinematic hardening.

ARGUMENT LIST

Type	Name		Description
integer	MODE	>	Initialisation/Switching mode.
integer	NLARGE	>	Large strain flag.
integer	NTYPE	>	Stress state type flag.
double precision	LALGVC	<>	Array of logical algorithmic variables at Gauss point. Current values.
double precision	LALGVL	<>	Array of logical algorithmic variables at Gauss point. Last converged (equilibrium) values.
double precision	RALGVC	<>	Array of real algorithmic variables at Gauss point. Current values.
double precision	RSTAVC	<>	Array of real state variables at Gauss point. Current values.
double precision	RSTAVL	<>	Array of real state variables at Gauss point. Last converged (equilibrium) values.
double precision	STRESC	<>	Array of stress components. Current values.
double precision	STRESL	<>	Array of stress components. Last converged (equilibrium) values.

SUBOUTINE TANGEN - MANUAL PAGE

DESCRIPTION
Sets prescribed displacements for arc-length tangential solution.

This routine sets the array of prescribed displacements as needed
for the tangential solution of the Arc-Length Method.

SUBOUTINE TUEL - MANUAL PAGE

DESCRIPTION
Thickness update for Hencky elastic model in plane stress

This routine updates the thickness for the (Hencky) elastic model under plane stress and large strains. It also computes the total deformation gradient (including the thickness strain contribution).

ARGUMENT LIST

Type	Name		Description
double precision	DETF	<	Determinant of the current total deformation gradient (including the thickness strain contribution).
double precision	RSTAVA	>	Array of current (updated) engineering logarithmic strain components.
double precision	THICK	<>	Initial (reference) thickness on entry. Returns as the current (updated) thickness.
integer	MODE	>	Flag. If MODE.NE.1, then only the total deformation gradient is computed. If MODE = 1, then the thickness is updated in addition.

SUBOUTINE TUVM - MANUAL PAGE

DESCRIPTION

Thickness update for large strain von Mises model in plane stress

This routine updates the thickness for the von Mises model under plane stress and large strains. It also computes the total deformation gradient (including the thickness strain contribution). The corresponding state update procedure for the von Mises model is carried out in subroutine SUVMPS.

ARGUMENT LIST

Type	Name		Description
double precision	DETF	<>	Determinant of the current in-plane deformation gradient on entry. Returns as the determinant of the current total deformation gradient (including the thickness strain contribution).
double precision	RSTAVA	>	Array of current (updated) real state variables other than the stress tensor components.
double precision	THICK	<>	Initial (reference) thickness on entry. Returns as the current (updated) thickness.
integer	MODE	>	Flag. If MODE.NE.1, then only the total deformation gradient is computed. If MODE = 1, then the thickness is updated in addition.

SUBOUTINE UPCONF - MANUAL PAGE

DESCRIPTION
Kinematic/geometric configuration update

Given the current iterative displacements, this routine updates the global arrays of incremental and total nodal displacements as well as the nodal coordinates. Nodal coordinates are updated only for large deformation analyses.