

# 3d residual stress project

## Nomenclature

$\alpha$	penalty factor for the symmetry-surface constraint
$\beta$	penalty factor for the traction-free constraint
$\delta$	Kronecker delta
$\kappa$	scaling constant for the ellipsoids
$\lambda$	penalty factor for the equilibrium constraint
$\epsilon$	lattice strains obtain by diffraction measurements
$\bar{\mathbf{0}}$	zero vector
$\bar{\mathbf{N}}^s$	vectorized form of surface shape functions for an element
$\bar{\mathbf{N}}$	vectorized form of shape functions for an element
$\Sigma(\mathbf{x})$	stress at position $\mathbf{x}$
$\sigma_{\mathbf{c}}(\mathbf{x})$	interpolated value of measured stress at any position $\mathbf{x}$
$\sigma_{\mathbf{c}}(\mathbf{x}_{\mathbf{c}})$	measured stress at position $\mathbf{x}_{\mathbf{c}}$
$\mathbf{a}(\mathbf{x})$	coefficients of monomial used in MLS method
$\mathbf{A}$	mapping between lattice strains and harmonic coefficients
$\mathbf{B}$	a special matrix formed by the derivative of shape functions
$\mathbf{C}^{\text{el}}$	elemental equilibrium constraint matrix
$\mathbf{C}_I$	global equilibrium constraint matrix
$\mathbf{D}^{\text{el}}$	elemental symmetry surface constraint matrix
$\mathbf{D}_I$	global symmetry surface constraint matrix
$\mathbf{F}^{\text{el}}$	elemental measured stress vector
$\mathbf{F}_I$	global measured stress vector
$\mathbf{G}$	mapping to get in-plane surface tractions from surface tractions
$\mathbf{I}$	identity matrix
$\mathbf{J}$	mapping that gives in-plane tractions from the stress components

$\mathbf{K}$	finite element constraints
$\mathbf{L}$	mapping between harmonic coefficients and orientation averaged stress
$\mathbf{M}^{\text{el}}$	elemental premultiplier of nodal stresses
$\mathbf{M}_I$	global (assembled) premultiplier of nodal stresses
$\mathbf{N}^s$	matrix formed by the shape functions for the surface nodes of an element $\forall$ dof
$\mathbf{N}$	matrix formed by the shape functions for the nodes of an element $\forall$ dof
$\mathbf{P}$	mapping to get tractions from the stress components
$\mathbf{p}$	monomial with a linear basis
$\mathbf{Q}$	mapping between orientation averaged stress and nodal stress vector
$\mathbf{R}$	residual
$\mathbf{r}$	surface normal
$\mathbf{s}^s$	stress vector containing all the nodes of the surface of an element
$\mathbf{s}$	stress vector containing all the nodes of an element
$\mathbf{S}_I$	global stress vector at the global node I in the mesh
$\mathbf{s}_I$	elemental stress vector at the global node I in the mesh
$\mathbf{t}(\mathbf{x})$	surface traction vector
$\mathbf{T}^{\text{el}}$	elemental traction-free surface constraint matrix
$\mathbf{T}_I$	global traction-free surface constraint matrix
$\mathbf{Y}$	a mapping used to construct interpolation function used in MLS method
$\mathbf{Z}$	another mapping used to construct interpolation function used in MLS method
$\Pi$	volume of the complete domain
$\Psi$	weight of the residual
$\rho$	constant radius of spheres used in MLS method
$\xi, \eta, \zeta$	natural (mapped) coordinates of elements
$A$	area of an element
$J$	square norm of the residual used in MLS method
$n$	number of nodes per element
$N_I^s$	surface shape functions of node I
$N_I$	shape functions of node I
$ndv$	number of diffraction measurement locations
$ns$	number of surface nodes per element
$q$	distance from the position of measurement used in MLS method

$s_I^{ij}$	ij component of stress at the node I
$V$	volume of an element
$w$	weight used in MLS method
$x, y, z$	coordinate positions

The object of this analysis is to find the stress distribution in a body (i.e. machine component, sample after a forming process) using lattice strain measurements. That is called residual stress but it should not be confused with the residual used in the analysis.

Residual refers to the difference between the measured orientation averaged stresses that are measured from diffraction experiments and the macroscopic stress distribution which are obtained from the nodal values of stress. Body under interest is discretized with finite elements. The objective is to find the nodal values of stresses that minimize the residual<sup>1</sup>,  $\mathbf{R}$ , weakly (in an integral form) as in eq. (1).

In eq. (1), the difference between the macroscopic stress,  $\Sigma(\mathbf{x})$ , at position  $\mathbf{x}$ , and the stress measured,  $\sigma_{\mathbf{c}}$ , at position at  $\mathbf{x}_{\mathbf{c}}$ , is minimized over the domain  $\Pi$ .  $\Psi$  is the weight of the residual. The measured stress,  $\sigma_{\mathbf{c}}$ , is an orientation averaged quantity and it is obtained from the lattice strain measurements (More details are discussed in section 6).

$$\mathbf{R} = \int_{\Pi} \Psi [\Sigma(\mathbf{x}) - \delta(\mathbf{x} - \mathbf{x}_{\mathbf{c}}) \sigma_{\mathbf{c}}] d\Pi \quad (1)$$

Later on, several other physical constraints acting on the body,  $\Pi$ , (i.e. equilibrium, traction free surfaces) will be introduced to the residual.

## 1 residual

Residual is indicated with  $\mathbf{R}$  in eq. (1) which comprises of two main integrals of 1.1 macroscopic stress (to be calculated from the nodal stresses) and, 1.2 measured stress vector.

### 1.1 macroscopic stress

Eq. (2) shows an example use of interpolation functions to find the magnitude of  $zz$  component of stress,  $\Sigma(x)^{zz}$ , in an element at position  $\mathbf{x}$  using the nodal values of stress  $s_I^{zz}$  and the shape function of that element  $N_I(\mathbf{x})$ . The subscript  $I$  refers to the node number of that element. Shape functions of different elements are shown in appendix A.

$$\Sigma^{zz}(x) = \sum_I N_I(\mathbf{x}) s_I^{zz} \quad (2)$$

For a general case, stress has 6 degrees of freedom (dof) that are independent of each other. Hence, to interpolate all 6 components of stress, we write eq. (2) for each stress component as a matrix equation, (3).

$$\Sigma(\mathbf{x}) = \mathbf{N}(\mathbf{x}) \mathbf{s} \quad (3)$$

$\mathbf{s}$  is the vector that consists of nodal stress vectors,  $\mathbf{s}_I$ , as shown in eq. (4) where  $n$  number of nodes per element. The superscript  $T$  is used to indicate transpose.

$$\mathbf{s} = [\mathbf{s}_1 \mathbf{s}_2 \cdots \mathbf{s}_n]^T \quad (4)$$

---

<sup>1</sup>Bold fonts are used indicate matrices and vectors.

$\mathbf{s}_I$  is the vectorized form of stress at each node  $I$  as in eq. (5).

$$\mathbf{s}_I = \begin{bmatrix} s_I^{xx} \\ s_I^{yy} \\ s_I^{zz} \\ s_I^{xy} \\ s_I^{xz} \\ s_I^{yz} \end{bmatrix} \quad (5)$$

In eq. (6),  $\mathbf{N}$  is a matrix formed by the shape functions and by the zero vectors for 6 dof calculated at any position  $\mathbf{x}$ . Positions are not indicated anymore for simplicity.

$$\mathbf{N} = \begin{bmatrix} \bar{\mathbf{N}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} \\ \bar{\mathbf{0}} & \bar{\mathbf{N}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} \\ \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{N}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} \\ \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{N}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} \\ \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{N}} & \bar{\mathbf{0}} \\ \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{0}} & \bar{\mathbf{N}} \end{bmatrix} \quad (6)$$

$\bar{\mathbf{N}}$  stands for the row vector that contains the shape functions of  $n$  number of nodes per element.

$$\bar{\mathbf{N}} = [N_1 \ N_2 \ \cdots \ N_n]_{1 \times n} \quad (7)$$

Zero vector for  $n$  number of nodes per element,  $\bar{\mathbf{0}}$ , is as shown in eq. (8).

$$\bar{\mathbf{0}} = [0 \ 0 \ \cdots \ 0]_{1 \times n} \quad (8)$$

Mass-like matrix, that is the elemental pre-multiplier or coefficient matrix of the nodal stresses,  $\mathbf{M}^{\text{el}}$ , is given by the volume integral shown in (9). This integral is calculated for each element in the mesh using numerical quadrature.

$$\mathbf{M}^{\text{el}} = \int_V \mathbf{N}^T \mathbf{N} dV \quad (9)$$

Later, the elemental matrices,  $\mathbf{M}^{\text{el}}$ , will be assembled to a global mass-like matrix,  $\mathbf{M}_I$ .

## 1.2 measured stress vector

The measured stress,  $\sigma_c$ , is an orientation averaged quantity. An elemental force like vector,  $\mathbf{F}^{\text{el}}$ , for each of the element in the mesh must be calculated from the given or measured stresses, (10).

$$\mathbf{F}^{\text{el}} = \int_V \mathbf{N}^T \sigma_c dV \quad (10)$$

The integral is estimated using numerical quadrature. Therefore, we need to know the values of stresses at the quadrature points. Here, we used two main approaches; i. constant stress and ii. moving least squares (MLS) method to find the values of stresses approximately at the quadrature points. The details of MLS method will be explained later. Currently, it is assumed that the magnitude of stress at the quadrature points are known. Therefore the integral in eq. (10) can be calculated to find the measured stress vector for each element in the finite element mesh. The elemental measured stress vector is then assembled to a global measured stress vector,  $\mathbf{F}_I$  by usual assembly procedure.

Eqs. (9) and (10) give the matrix equation that is used to find the nodal values of stress vector,  $\mathbf{S}_I$  (without any constraints is given in (11)).

$$\mathbf{M}_I \mathbf{S}_I = \mathbf{F}_I \quad (11)$$

## 2 equilibrium constraint

Equilibrium suggests the divergence of the stresses to vanish at any point inside the body on which no body and surface loads are acting and when body is not accelerating. Its being equal to a  $\mathbf{0}$  vector indicates equilibrium all three directions for three dimensions.

$$\int_V \nabla \cdot \boldsymbol{\Sigma}(\mathbf{x}) dV = \mathbf{0} \quad (12)$$

$\boldsymbol{\Sigma}(\mathbf{x})$  in eq. (12) is obtained by eq. (3) using shape functions and nodal values of stress,  $\mathbf{s}$ . Nodal stresses can be taken outside the integral expression hence,  $\mathbf{C}^{\text{el}}$  refers to the matrix that premultiplies the nodal stresses and to calculate equilibrium. Here, we set it to be zero.

$$\mathbf{C}^{\text{el}} \mathbf{s} = \mathbf{0} \quad (13)$$

Equilibrium condition reveals elemental constraint matrix  $\mathbf{C}^{\text{el}}$  (14) in which  $\mathbf{B}$  is a special matrix formed by the derivative of shape functions.

$$\mathbf{C}^{\text{el}} = \int_V \mathbf{B}^T \mathbf{B} dV \quad (14)$$

The dimension of  $\mathbf{B}$  is 3 by 6 times n (6 dof times the number of nodes per element). The subscripts stand for the node numbers.

$$\mathbf{B} = [\mathbf{B}_1 \mathbf{B}_2 \cdots \mathbf{B}_n]_{3 \text{ by } 6 \times n} \quad (15)$$

(16) shows an example calculation of  $\mathbf{B}_1$  for the 1st node of an element.

$$\mathbf{B}_1 = \begin{bmatrix} \frac{dN_1}{dx} & 0 & 0 & \frac{dN_1}{dy} & \frac{dN_1}{dz} & 0 \\ 0 & \frac{dN_1}{dy} & 0 & \frac{dN_1}{dx} & 0 & \frac{dN_1}{dz} \\ 0 & 0 & \frac{dN_1}{dz} & 0 & \frac{dN_1}{dx} & \frac{dN_1}{dy} \end{bmatrix} \quad (16)$$

The integral in (14) is calculated using numerical quadrature to obtain the elemental constraint,  $\mathbf{C}_{\text{el}}$ . The constraints for each element is assembled to obtain the global equilibrium constraint matrix,  $\mathbf{C}_I$ .

## 3 traction free surface constraint

The tractions on the free surfaces,  $\mathbf{t}(\mathbf{x})$ , must vanish. In (17),  $\mathbf{r}(\mathbf{x})$  is the surface normal and  $\boldsymbol{\Sigma}(\mathbf{x})$  is the stress at any point on that surface.

$$\mathbf{t}(\mathbf{x}) = \int_A \mathbf{r}(\mathbf{x}) \cdot \boldsymbol{\Sigma}(\mathbf{x}) dA = \mathbf{0} \quad (17)$$

The traction free surface constraint is imposed by the use of shape functions of the surfaces of the elements. The stresses on the surface of an element,  $\boldsymbol{\Sigma}(\mathbf{x})$ , is given by the surface shape functions,  $\mathbf{N}^s(\mathbf{x})$ , and by the stress components at the nodes of that surface,  $\mathbf{s}^s$  as shown in eq. (18). Please note that the shape functions,  $\mathbf{N}^s(\mathbf{x})$ , are one dimension less than the shape functions of the elements,  $\mathbf{N}(\mathbf{x})$ .

$$\boldsymbol{\Sigma}(\mathbf{x}) = \mathbf{N}^s(\mathbf{x}) \mathbf{s}^s \quad (18)$$

$\mathbf{s}^s$  is the vector that consists of nodal stress vectors of the surfaces,  $\mathbf{s}_I$ , as shown in eq. (19) where  $ns$  number of nodes per surface.

$$\mathbf{s}^s = [\mathbf{s}_1 \mathbf{s}_2 \cdots \mathbf{s}_{ns}]^T \quad (19)$$

$s_I$  in which I stands for the node numbers of the surface under concern.

$$s_I = \begin{bmatrix} s_I^{xx} \\ s_I^{yy} \\ s_I^{zz} \\ s_I^{xy} \\ s_I^{xz} \\ s_I^{yz} \end{bmatrix} \quad (20)$$

$\mathbf{N}^s$  is the matrix that contains the shape functions to find the stress on the surface of an element, eq. (21).

$$\mathbf{N}^s = \begin{bmatrix} \bar{\mathbf{N}}^s & \bar{0} & \bar{0} & \bar{0} & \bar{0} & \bar{0} \\ \bar{0} & \bar{\mathbf{N}}^s & \bar{0} & \bar{0} & \bar{0} & \bar{0} \\ \bar{0} & \bar{0} & \bar{\mathbf{N}}^s & \bar{0} & \bar{0} & \bar{0} \\ \bar{0} & \bar{0} & \bar{0} & \bar{\mathbf{N}}^s & \bar{0} & \bar{0} \\ \bar{0} & \bar{0} & \bar{0} & \bar{0} & \bar{\mathbf{N}}^s & \bar{0} \\ \bar{0} & \bar{0} & \bar{0} & \bar{0} & \bar{0} & \bar{\mathbf{N}}^s \end{bmatrix} \quad (21)$$

where  $\bar{\mathbf{N}}^s$  is the row vector that contains the shape functions of the surface nodes.  $ns$  stands for the number of the nodes at the surface of an element.

$$\bar{\mathbf{N}}^s = [N_1^s N_2^s \cdots N_{ns}^s]_{1 \text{ by } ns} \quad (22)$$

The surface tractions from the stresses are calculated by projecting the stresses onto the surface normal  $\mathbf{r}$  using Cauchy relation.  $\mathbf{P}$  matrix that contains the surface normals to map the correct stress component to the desired traction vector component. For example, vector  $r_1$  stands for the surface normal vector component along the 1st direction (subscripts of  $\mathbf{r}$  vector denote the components along the three principal directions).

$$\mathbf{P} = \begin{bmatrix} r_1 & 0 & 0 & r_2 & r_3 & 0 \\ 0 & r_2 & 0 & r_1 & 0 & r_3 \\ 0 & 0 & r_3 & 0 & r_1 & r_2 \end{bmatrix} \quad (23)$$

The shape functions are used to weight the free surface constraint, eq. (24). By taking the nodal stresses out of the integral, the free surface constraint gives;

$$\mathbf{T}^{el} = \int_A \mathbf{N}^s \mathbf{T} \mathbf{P}^T \mathbf{P} \mathbf{N}^s dA \quad (24)$$

The surface normal,  $\mathbf{r}$ , at the quadrature points is calculated using the two in-plane directional derivatives at the quadrature points; by taking the cross product of these directional derivatives that are tangent to the surface at the quadrature points.

The integral in eq. 24 is calculated using numerical quadrature. The elemental free surface constraints,  $\mathbf{T}^{el}$ , are assembled to the corresponding nodes in the global constraint matrix,  $\mathbf{T}_I$ .

## 4 symmetry surface constraint

Symmetry requires the in plane surface tractions to vanish. Surface traction,  $\mathbf{t}(\mathbf{x})$ , is calculated using eq. (17).

$$\int_A (\mathbf{I} - \mathbf{r}(\mathbf{x}) \otimes \mathbf{r}(\mathbf{x})) \cdot \mathbf{t}(\mathbf{x}) dA = \mathbf{0} \quad (25)$$

The application of this constraint is similar to the free surface constraint. In addition to the  $\mathbf{P}$  matrix in eq. (23), a different matrix,  $\mathbf{G}$  is necessary to get the in plane tractions from the surface tractions.

$$\mathbf{G} = \mathbf{I} - \mathbf{r} \otimes \mathbf{r} \quad (26)$$

$$\mathbf{G} = \begin{bmatrix} 1 - r_1 r_1 & -r_1 r_2 & -r_1 r_3 \\ -r_2 r_1 & 1 - r_2 r_2 & -r_2 r_3 \\ -r_3 r_1 & -r_3 r_2 & 1 - r_3 r_3 \end{bmatrix} \quad (27)$$

The symmetry surface constraint becomes as shown in eq. (28) after taking out the nodal stresses from the integral;

$$\mathbf{D}^{\text{el}} = \int_A \mathbf{N}^{\text{s T}} \mathbf{J}^{\text{T}} \mathbf{J} \mathbf{N}^{\text{s}} dA = \mathbf{0} \quad (28)$$

where

$$\mathbf{J} = \mathbf{G} \mathbf{P} \quad (29)$$

The elemental symmetry surface constraints,  $\mathbf{D}^{\text{el}}$ , are then assembled to the corresponding nodes of the global constraint matrix,  $\mathbf{D}_I$ .

## 5 total residual together with constraints

The equation to find the nodal stresses together with the free surface and the equilibrium constraints becomes as shown in eq. (30).  $\mathbf{M}_I$  is the mass-like matrix that originates from the residual shown in eq. (1).  $\lambda$  is the factor that penalizes the equilibrium constraint,  $\mathbf{C}_I$ .  $\mathbf{T}_I$  is the constraint for the traction-free surfaces and  $\beta$  is the corresponding penalty factor.  $\mathbf{D}_I$  is the constraint for the symmetry surfaces and  $\alpha$  is the corresponding penalty factor.  $\mathbf{F}_I$  is the global force vector obtained after assembly of elemental force vectors.

$$[\mathbf{M}_I + \lambda \mathbf{C}_I + (\beta \mathbf{T}_I) + (\alpha \mathbf{D}_I)] \mathbf{S}_I = \mathbf{F}_I \quad (30)$$

A simpler representation of (30) is shown in (31) in which  $\mathbf{K}$  is the total mass-like matrix.

$$\mathbf{K} \mathbf{S} = \mathbf{F} \quad (31)$$

Please note that in eq. (30), the surface constraints are imposed only at the corresponding surface nodes (That is why there are shown in parenthesis).

## 6 finite element constraints with strain pole figure inversion

Strain pole figure inversion does not have a unique solution. Therefore, equilibrium and boundary constraints is reformulated to solve for the stress distribution simultaneously together with the strain pole figure inversion as shown in eq. (32).

In eq. (32),  $\mathbf{K}$  represents the finite element constraints (premultiplier of nodal stress in (30)).  $\mathbf{S}$  is the nodal stress vector of the mesh.  $\mathbf{L}$  is used to find the orientation averaged stress vector from the harmonic coefficients of the lattice strain distribution function.  $\mathbf{Q}$  converts the orientation averaged stress to nodal stress vectors. Formulation of  $\mathbf{Q}$  is dependent on the method used to find the stresses at numerical quadrature locations from the diffraction data points (i.e. assuming constant stress over the element, MLS method).  $\mathbf{A}$  is the mapping between harmonic coefficients of the lattice strain distribution function,  $\mathbf{H}$ , and the lattice strains,  $\epsilon$ .

$$\begin{bmatrix} \mathbf{K} & -\mathbf{Q} \mathbf{L} \\ \mathbf{0} & \mathbf{A} \end{bmatrix} \begin{bmatrix} \mathbf{S} \\ \mathbf{H} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \epsilon \end{bmatrix} \quad (32)$$

Nodal force-like vector is given as;

$$\mathbf{F} = \mathbf{Q} \mathbf{L} \mathbf{H} \quad (33)$$

## 7 moving least square (MLS) method

Moving least square method is used to interpolate the stresses at the quadrature points using the diffraction data that is collected at the diffraction data points. It is done in order to calculate the force-like vector shown in the integral in (10). This way we can take out the dependency of measurement to the mesh, fig. 1.

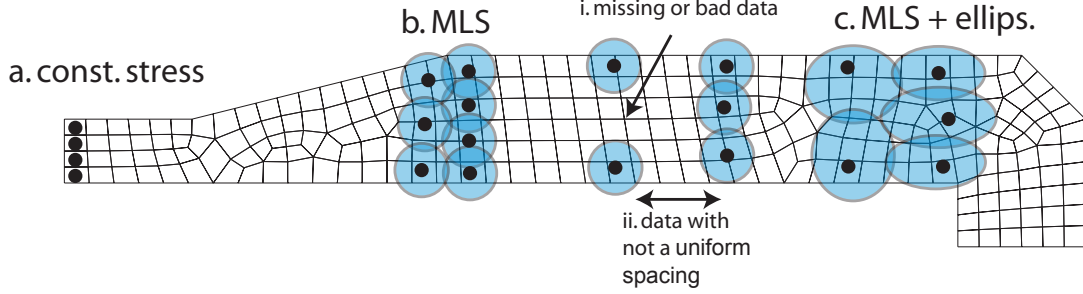


Figure 1: An example mesh showing different approaches to find the measured stress vector. **a.** constant stress assignment to quadrature points requires measured data inside each and every finite element of the body (or the mesh should adjusted to be such.), **b.** MLS method using spheres with constant radii, **c.** MLS method using ellipsoids with various sizes.

$\sigma_c(\mathbf{x})$  stands for the interpolated value of measured stress at any position  $\mathbf{x}$ . We want to find the values of stresses at the quadrature point locations.  $\sigma_c$  refers to the measured stress at the diffraction measurement location  $\mathbf{x}_c$ .

MLS method suggests the use of interpolation function,  $\mathbf{p}(\mathbf{x})$ , with coefficients of  $\mathbf{a}(\mathbf{x})$ . The variable of interest, in our case it is the interpolated measured stress,  $\sigma_c(\mathbf{x})$ , can be calculated at any position if we know the coefficients,  $\mathbf{a}(\mathbf{x})$ , of the polynomial bases,  $\mathbf{p}(\mathbf{x})$ . Eq. (34) performs this interpolation;

$$\sigma_c(\mathbf{x}) = \mathbf{p}^T(\mathbf{x}) \cdot \mathbf{a}(\mathbf{x}) \quad (34)$$

where  $\mathbf{p}(\mathbf{x})$  is the monomial with a linear basis.

$$\mathbf{p}^T(\mathbf{x}) = [1 \ x \ y \ z] \quad (35)$$

and  $\mathbf{a}(\mathbf{x})$  is the coefficients of that monomial. The square norm of the residual (This residual should not be confused with the residual that is minimized in a weak sense over the finite element mesh and, also with the residual stresses.) becomes as shown in eq. (36) in which  $w$  is the weight at location  $x$ . Summation  $\sum_c$  indicates sum over the diffraction measurements.  $ndv$  indicates number of diffraction measurements.

$$J = \sum_c^{ndv} w(\mathbf{x} - \mathbf{x}_c) [\mathbf{p}^T(\mathbf{x}_c) \cdot \mathbf{a}(\mathbf{x}_c) - \sigma_c]^2 \quad (36)$$

We want the coefficients that minimize the difference between the measured and interpolated values at the position of measurement, residual. The polynomial coefficients  $\mathbf{a}(\mathbf{x})$  that minimize  $J$  is shown in eq. (37).

$$\mathbf{a}(\mathbf{x}) = \mathbf{Y}(\mathbf{x})^{-1} \mathbf{Z}(\mathbf{x}) \sigma_c \quad (37)$$

where

$$\mathbf{Y}(\mathbf{x}) = \sum_{c=1}^{ndv} w_c(\mathbf{x} - \mathbf{x}_c) \mathbf{p}^T(\mathbf{x}_c) \mathbf{p}(\mathbf{x}_c) \quad (38)$$

$$\mathbf{Z}(\mathbf{x}) = [ \ w_1(\mathbf{x} - \mathbf{x}_1) \mathbf{p}(\mathbf{x}_1) \quad w_2(\mathbf{x} - \mathbf{x}_2) \mathbf{p}(\mathbf{x}_2) \quad \cdots \quad w_{ndv}(\mathbf{x} - \mathbf{x}_{ndv}) \mathbf{p}(\mathbf{x}_{ndv}) \ ] \quad (39)$$



$$\sigma_{\mathbf{c}}^{\mathbf{T}} = [ \sigma_1 \quad \sigma_2 \quad \cdots \quad \sigma_{ndv} ] \quad (40)$$

The method is called moving least squares because of the dependence of weight on the position  $\mathbf{x}$ . This way local character of the measurement is preserved better compared to weighted least square or least square methods. The quartic splines are used as weighting functions.

$$w(q) = \begin{cases} 1 - 6q^2 + 8q^3 - 3q^4 & \text{if } q < 1 \\ 0 & \text{otherwise} \end{cases} \quad (41)$$

$$q = \frac{\|\mathbf{x} - \mathbf{x}_{\mathbf{c}}\|}{\rho} \quad (42)$$

$\rho$  is the radius of the diffraction volume for a sphere diffraction volume, fig. 1. Further details of the method can be found in the ref. [1].

## 7.1 MLS with ellipsoids

The use of spheres with constant radii may cause loss of information. When spheres are used instead of ellipsoids, the user must choose a very large radius to cover the whole disk hence this may cause more information to be lost. Therefore ellipsoids with varying stretch ratios and orientations are constructed around the diffraction volumes from the given diffraction volume coordinates.

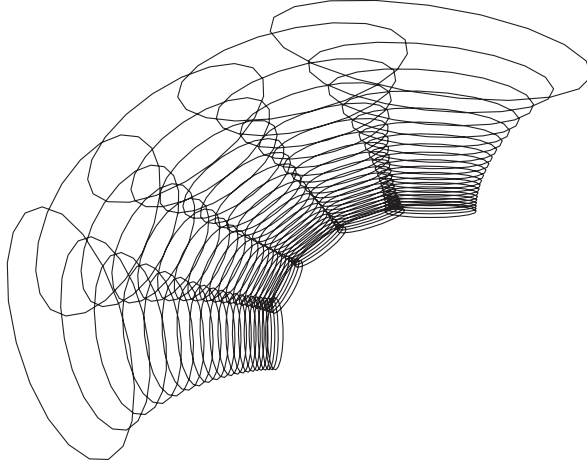


Figure 2: The ellipses that are constructed for a 2D example of a shrink-fit disk problem.

Fig. 2 shows the ellipses that are constructed for a 2D example of a shrink-fit disk problem using the diffraction volume position coordinates.

The MLS method used for the ellipsoids is the same, only the way that the variable  $q$  in eq. (41) is computed differently for the ellipsoids than the spheres. First, the parametric equations for the ellipsoids and their orientation are determined. Second, the position under interest is transformed into reference frame of the ellipsoid. Third, the position coordinates of the point (the point at which the weight  $w(q)$  is desired - i.e. quadrature points.) is inserted to the equation of the ellipsoid to find the function value of that point.

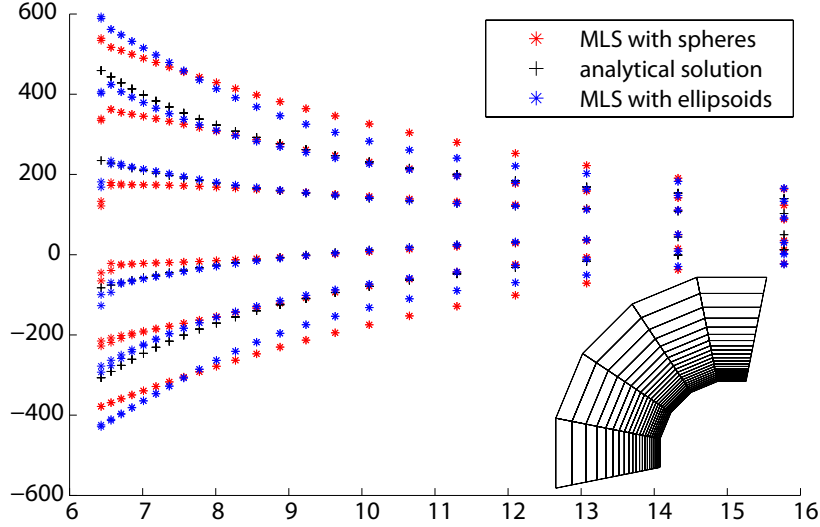


Figure 3:  $\Sigma^{xx}$  obtained using the analytical solution for the stress at the diffraction data points using spheres vs. ellipsoids and the expected analytical solution. Smallest possible sizes both for the spheres and the ellipsoids are used for comparison.

That point would have a function value of unity at the surface of an ellipsoid. Therefore, finally, unity minus the function value at that point is used as the variable  $q$  in eq. (41) for ellipsoids.

The user may need to rescale the size of the ellipsoids since it may not cover a quadrature point in the mesh. Therefore a scaling constant,  $\kappa$ , is introduced to adjust the size of the ellipsoids.

Fig. 3, shows the results obtained using the analytical solution for the stress at the diffraction data points. The use of spheres vs. ellipsoids in conjunction with the MLS method are shown and compared the expected analytical solution at the nodes of the finite element mesh. The use of spheres with constant radii causes loss of information especially at the locations where the diffraction data points tightly spaced as expected. The solution with the ellipsoids gets closer to the expected behavior (analytical solution) in comparison to the results obtained by the use of spheres.

## A element library

### A.1 8-node brick element

Figure 4 shows the node numbering used for the volume and surfaces of 8 node brick elements. The order of the nodes of a face of an element must be defined using the nodal ordering<sup>2</sup> shown in fig. 4.

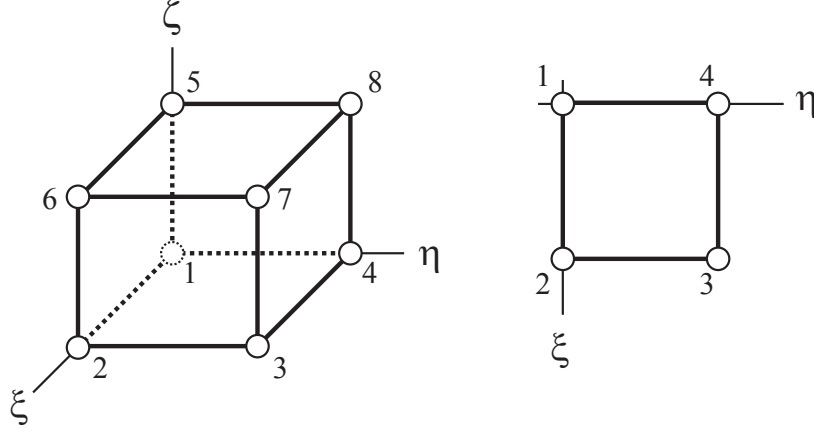


Figure 4: Node numbering of the volume and the surfaces of an 8 node brick element.

The shape functions of 8 node brick elements are given as below. Note, the natural coordinates  $(\xi, \eta, \zeta)$  range from 0 to +1.

$$N_1 = (1 - \xi)(1 - \eta)(1 - \zeta) \quad (43)$$

$$N_2 = \xi(1 - \eta)(1 - \zeta) \quad (44)$$

$$N_3 = \xi\eta(1 - \zeta) \quad (45)$$

$$N_4 = (1 - \xi)\eta(1 - \zeta) \quad (46)$$

$$N_5 = (1 - \xi)(1 - \eta)\zeta \quad (47)$$

$$N_6 = \xi(1 - \eta)\zeta \quad (48)$$

$$N_7 = \xi\eta\zeta \quad (49)$$

$$N_8 = (1 - \xi)\eta\zeta \quad (50)$$

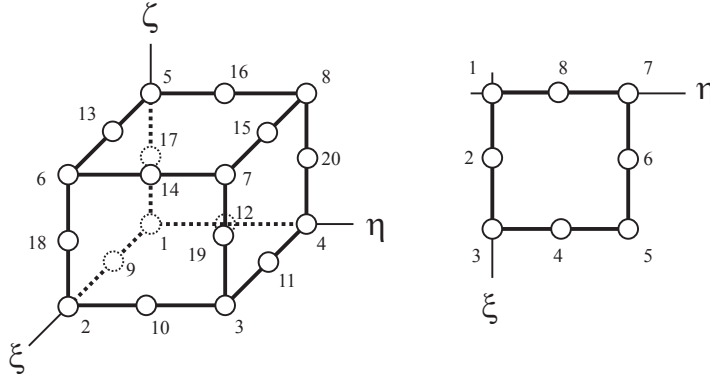


Figure 5: Node numbering of the volume and the surfaces of an 20 node brick element.

## A.2 20-node brick element

$$N_1 = 2(1 - \xi)(1 - \eta)(1 - \zeta)(0.5 - \xi - \eta - \zeta) \quad (51)$$

$$N_2 = -2\xi(1 - \eta)(1 - \zeta)(0.5 - \xi + \eta + \zeta) \quad (52)$$

$$N_3 = 2\xi\eta(1 - \zeta)(\xi + \eta - \zeta - 1.5) \quad (53)$$

$$N_4 = -2(1 - \xi)\eta(1 - \zeta)(0.5 + \xi - \eta + \zeta) \quad (54)$$

$$N_5 = -2(1 - \xi)(1 - \eta)\zeta(0.5 + \xi + \eta - \zeta) \quad (55)$$

$$N_6 = -2\xi(1 - \eta)\zeta(1.5 - \xi + \eta - \zeta) \quad (56)$$

$$N_7 = -2\xi\eta\zeta(2.5 - \xi - \eta - \zeta) \quad (57)$$

$$N_8 = -2(1 - \xi)\eta\zeta(1.5 + \xi - \eta - \zeta) \quad (58)$$

$$N_9 = 4\xi(1 - \xi)(1 - \eta)(1 - \zeta) \quad (59)$$

$$N_{10} = 4\xi\eta(1 - \eta)(1 - \zeta) \quad (60)$$

$$N_{11} = 4\xi(1 - \xi)\eta(1 - \zeta) \quad (61)$$

$$N_{12} = 4(1 - \xi)\eta(1 - \eta)(1 - \zeta) \quad (62)$$

$$N_{13} = 4\xi(1 - \xi)(1 - \eta)\zeta \quad (63)$$

$$N_{14} = 4\xi\eta(1 - \eta)\zeta \quad (64)$$

$$N_{15} = 4\xi(1 - \xi)\eta\zeta \quad (65)$$

$$N_{16} = 4(1 - \xi)\eta(1 - \eta)\zeta \quad (66)$$

$$N_{17} = 4(1 - \xi)(1 - \eta)\zeta(1 - \zeta) \quad (67)$$

$$N_{18} = 4\xi(1 - \eta)\zeta(1 - \zeta) \quad (68)$$

$$N_{19} = 4\xi\eta\zeta(1 - \zeta) \quad (69)$$

$$N_{20} = 4(1 - \xi)\eta\zeta(1 - \zeta) \quad (69)$$

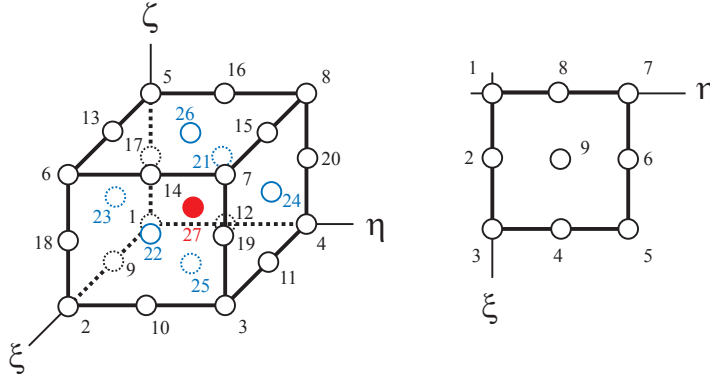


Figure 6: Node numbering of the volume and the surfaces of an 27 node brick element.

### A.3 27-node brick element

$$N_1 = (1 - 2\xi)(1 - \xi)(1 - 2\eta)(1 - \eta)(1 - 2\zeta)(1 - \zeta) \quad (70)$$

$$N_2 = -(1 - 2\xi)\xi(1 - 2\eta)(1 - \eta)(1 - 2\zeta)(1 - \zeta) \quad (71)$$

$$N_3 = (1 - 2\xi)\xi(1 - 2\eta)\eta(1 - 2\zeta)(1 - \zeta) \quad (72)$$

$$N_4 = -(1 - 2\xi)(1 - \xi)(1 - 2\eta)\eta(1 - 2\zeta)(1 - \zeta) \quad (73)$$

$$N_5 = -(1 - 2\xi)(1 - \xi)(1 - 2\eta)(1 - \eta)(1 - 2\zeta)\zeta \quad (74)$$

$$N_6 = (1 - 2\xi)\xi(1 - 2\eta)(1 - \eta)(1 - 2\zeta)\zeta \quad (75)$$

$$N_7 = -(1 - 2\xi)\xi(1 - 2\eta)\eta(1 - 2\zeta)\zeta \quad (76)$$

$$N_8 = (1 - 2\xi)(1 - \xi)(1 - 2\eta)\eta(1 - 2\zeta)\zeta \quad (77)$$

$$N_9 = 4\xi(1 - \xi)(1 - 2\eta)(1 - \eta)(1 - 2\zeta)(1 - \zeta) \quad (78)$$

$$N_{10} = -4(1 - 2\xi)\xi\eta(1 - \eta)(1 - 2\zeta)(1 - \zeta) \quad (79)$$

$$N_{11} = -4\xi(1 - \xi)(1 - 2\eta)\eta(1 - 2\zeta)(1 - \zeta) \quad (80)$$

$$N_{12} = 4(1 - 2\xi)(1 - \xi)\eta(1 - \eta)(1 - 2\zeta)(1 - \zeta) \quad (81)$$

$$N_{13} = -4\xi(1 - \xi)(1 - 2\eta)(1 - \eta)(1 - 2\zeta)\zeta \quad (82)$$

$$N_{14} = 4(1 - 2\xi)\xi\eta(1 - \eta)(1 - 2\zeta)\zeta \quad (83)$$

$$N_{15} = 4\xi(1 - \xi)(1 - 2\eta)\eta(1 - 2\zeta)\zeta \quad (84)$$

$$N_{16} = -4(1 - 2\xi)(1 - \xi)\eta(1 - \eta)(1 - 2\zeta)\zeta \quad (85)$$

$$N_{17} = 4(1 - 2\xi)(1 - \xi)(1 - 2\eta)(1 - \eta)\zeta(1 - \zeta) \quad (86)$$

$$N_{18} = -4(1 - 2\xi)\xi(1 - 2\eta)(1 - \eta)\zeta(1 - \zeta) \quad (87)$$

$$N_{19} = 4(1 - 2\xi)\xi(1 - 2\eta)\eta\zeta(1 - \zeta) \quad (88)$$

$$N_{20} = -4(1 - 2\xi)(1 - \xi)(1 - 2\eta)\eta\zeta(1 - \zeta) \quad (89)$$

$$N_{21} = 16(1 - 2\xi)(1 - \xi)\eta(1 - \eta)\zeta(1 - \zeta) \quad (90)$$

$$N_{22} = -16(1 - 2\xi)\xi\eta(1 - \eta)\zeta(1 - \zeta) \quad (91)$$

$$N_{23} = -16\xi(1 - \xi)(1 - 2\eta)\eta\zeta(1 - \zeta) \quad (92)$$

$$N_{24} = 16\xi(1 - \xi)(1 - 2\eta)(1 - \eta)\zeta(1 - \zeta) \quad (93)$$

$$N_{25} = 16\xi(1 - \xi)\eta(1 - \eta)(1 - 2\zeta)(1 - \zeta) \quad (94)$$

$$(95)$$

<sup>2</sup>Nodal ordering of a face of an element must be such that the surface normal of an element must point outward direction.

$$N_{26} = -16\xi(1-\xi)\eta(1-\eta)(1-2\zeta)\zeta \quad (96)$$

$$N_{27} = 64\xi(1-\xi)\eta(1-\eta)\zeta(1-\zeta) \quad (97)$$

#### A.4 10-node tetragonal element

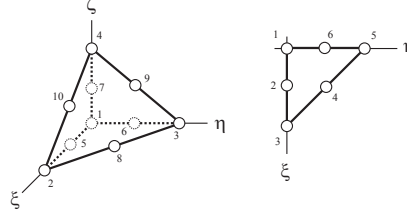


Figure 7: Node numbering of the volume and the surfaces of an 10 node tetragonal element.

$$N_1 = (1 - 2\xi - 2\eta - 2\zeta)(1 - \xi - \eta - \zeta) \quad (98)$$

$$N_2 = (2\xi - 1)\xi \quad (99)$$

$$N_3 = (2\eta - 1)\eta \quad (100)$$

$$N_4 = (2\zeta - 1)\zeta \quad (101)$$

$$N_5 = 4\xi(1 - \xi - \eta - \zeta) \quad (102)$$

$$N_6 = 4\eta(1 - \xi - \eta - \zeta) \quad (103)$$

$$N_7 = 4\zeta(1 - \xi - \eta - \zeta) \quad (104)$$

$$N_8 = 4\xi\eta \quad (105)$$

$$N_9 = 4\eta\zeta \quad (106)$$

$$N_{10} = 4\xi\zeta \quad (107)$$

$$(108)$$

## B inputs to the code

The inputs used in the analysis:

- **numel**: total number of elements in the mesh
- **numnp**: total number of nodes in the mesh
- **meltyp**: element type (8: 8 node brick, 20: 20 node brick, 27: 27 node brick, 10: 10 node tetrahedra)
- **x,y,z**: node coordinates
- **np**: element connectivity matrix
- **nps**: node numbers of each traction free surface (IMPORTANT: Node ordering must be such that counter-clockwise rotation must give the surface normal pointing OUTWARDS!)
- **npsyms**: node numbers of each symmetric surface (node ordering is important for the same reason as the traction free surfaces!)

- $\lambda$  : penalty factor for equilibrium constraint
- $\alpha$  : penalty factor for symmetry surface constraint
- $\beta$  : penalty factor for traction free surface constraint
- $\kappa$  : scaling constant to adjust the size of the ellipsoids

The inputs for the mesh are given in the file mesh\_data.m. The other inputs are declared in the main m-file; Residual\_Stress\_3d.m.

## C example: shrink fit disc

The analysis is applied to a portion of a disk that is shrink fitted to a shaft. The magnitudes of stresses at the quadrature points are found using the analytical solution to the problem. The surface constraints are shown in fig. 8.

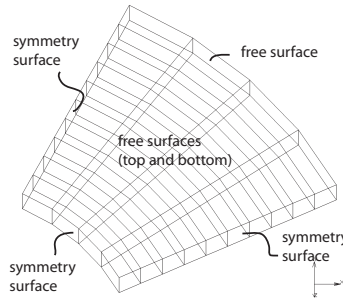


Figure 8: Surface constraints of the shrink fit disc problem.

The symmetry constraints allow us to study only a portion of disc. The stress distribution is obtained for the surface constraints together with the equilibrium constraint over the whole disc, fig. 9.

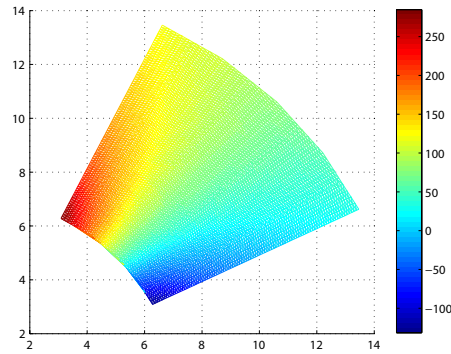


Figure 9: Results of the 3d analysis:  $\Sigma(x)^{xx}$  at the bottom surface of the disc.

## References

- [1] Beissel, S., Belytschko, T., 1996. Nodal integration of the element-free Galerkin method. Comp. Met. in App. Mech. and Eng. 139, 49-74.