Error Estimation for Thin Sheet Metal Forming Processes

Project Title	Modelling of Sheet Metal Forming
Project Number	ME97033
Cluster	Modelling of Sheet Metal Forming
Author	**A. Selman** University of Twente Netherlands Institute *for* Metals Research
Project Leaders	**J. Huetink and A.H. van den Boogaard**
Cluster co-ordinator	**J. Huetink**
Research Assistant	**T. Meinders**
Reference	WB.99 / NIMR-0154

April '99 / Revised Sept. '99

Notes on Error Estimation for Thin Sheet Metal Forming Processes

1. General Algorithm for Mesh Adaption in Metal Forming Processes

1. Generate an initial grid to represent the computational domain and to allow an adequate initial solution.

2. Advance the solution for an **adapted** number of steps.

3. Use the error indicator to determine whether mesh refinement is necessary. If yes, compute a new mesh distribution and continue - otherwise go to step 2.

4. Proceed with the mesh refinement and obtain the field values of the solution on the new grid by direct interpolation from the previous grid.

5. If the desired load interval has elapsed stop – otherwise go to step 2.

2. Evaluation of the thickness error

First, an area weighted nodal averaging technique is used to smooth the thickness as

$$\bar{h} = \frac{1}{\sum_{\kappa} A(T_{\kappa}(n))} \sum_{\kappa} A(T_{\kappa}(n))\ h(T_{\kappa}(n))$$

where $A(T_{\kappa}(n))$ and $h(T_{\kappa}(n)$ are the area and the thickness of element k containing node n, noted $T_{\kappa}(n)$.

Then, the thickness error is evaluated at 3 Gauss points using the following formula

$$e_h = \frac{1}{A_T} \int_{A_T} \left| \frac{h - \bar{h}}{h} \right| dA_T$$

Without a theoretical justification, experience has shown that a quadratic relation between this error measure and the element size gives satisfactory results. Therefore, we assume that

$$e_h = O(L^2\ )$$

where L is the actual element size (evaluated as the average of the element sides).

Henceforth, an element for which the size L and thickness error e_h are known, will be assigned the following element thickness size to satisfy a user specified tolerance η

$$L^h = L \sqrt{\frac{\eta}{e_h}}$$

3. Evaluation of the geometric error

First, a unique and element node numbering independent tangent set of axes is defined

$$n = g_3 = (n_x, n_y, n_z)$$
$$g_1 = (n_z + \frac{n_y^2}{a}, -\frac{n_x n_y}{a}, -n_x)$$
$$g_2 = (-\frac{n_x n_y}{a}, n_z + \frac{n_x^2}{a}, -n_y)$$

with $a = 1 + n_z$. If $a = 0$ $(n_x = n_y = 0,\ n_z = -1)$ then

$$g_1 = (1,0,0)$$
$$g_2 = (0,-1,0)$$

Given that linear triangles have been used exclusively, the tangent set is constant over each element. However, the variations of these sets from one element to its neighbours gives rise to a quadratic variation of the geometry that cannot be represented by facet elements.

To quantify the variations of the tangent sets of axes, a nodal smoothing technique is used as

$$\bar{g}_i(n) = \frac{1}{\sum_{\kappa} A(T_\kappa(n))} \sum_{\kappa} A(T_\kappa(n))\ g_i(T_\kappa(n))$$

where $A(T_\kappa(n))$ and $g_i(T_\kappa(n)$ are the area and vector i of the tangent set of axes of element k containing node n, noted $T_\kappa(n)$.

At a given node i, it is easy to calculate the smooth metric tensor

$$\bar{g}_{ij} = \bar{g}_i \cdot \bar{g}_j$$

As for the tangent sets, this variable varies linearly over each element. Given a metric tensor corresponding to the actual mesh and a smooth metric tensor corresponding to a higher order surface, the geometric error, as defined by Bonet [3], is obtained by first evaluating the deformation between the two surfaces. This is expressed simply in the surface plane, by the use of a pseudo Green Lagrange tensor E as

$$E_{ij} = \frac{1}{2}(\bar{g}_{ij} - g_{ij}) \qquad i, j = 1,2$$

The contravariant components of E is expressed as

$$E^{ij} = g^{ik} \, g^{jl} \, E_{kl}$$

The deformation in the normal direction, E_{33} is obtained by considering volume conservation

$$h \sqrt{g} = \bar{h} \sqrt{\bar{g}}$$

where g and $\bar{g}$ are the determinants of the metric tensor and smooth metric tensor, respectively. Consequently, E_{33} is given by

$$E_{33} = \frac{1}{2}\left[\frac{g}{\bar{g}} - 1\right]$$

The co- and contravariant components of the normal deformation are equal

$$E^{33} = E_{33}$$

An error norm measure of the error tensor is given by its invariant Π_E^2 as

$$\Pi_E^2 = E : E = E^{mn} E_{mn} + E^{33} E_{33}$$

Henceforth, the global error is evaluated by an integration of this field over the whole mesh

$$e_G^2 = \frac{1}{A} \int_A \Pi_E^2 \, dA$$

where A is the computational domain.

The local error over an element is given by

$$e_g^2 = \frac{1}{A_T} \int_{A_T} \Pi_E^2 \, dA_T$$

where A_T is the element area.

This error norm is converted into a geometric mesh size as

$$L^g = L \sqrt{\frac{\eta}{e_g}}$$

4. Evaluation of the new mesh size

The new mesh size is determined as

$$L_{new} = \max(L_{min}, min(L^h, L^g))$$

with $L_{min} = f \, . h(T)$ set to limit the minimum size allowed, with f a multiplying factor taking values in the range [2-4].

5. Remeshing conditions

The remeshing is started when either or both of the following conditions are verified

$$cond1 = \left| \frac{L - L_{new}}{L_{new}} \right|_{Mean} \geq tol1$$

$$cond2 = \left| \frac{L - L_{new}}{L_{new}} \right|_{Max} \geq tol2$$

If none of these conditions is verified (that is the actual mesh needs no refinement) then the solution is advanced for an adapted number of steps defined as

$$nsteps = min(nstep1, nstep2)$$

with

$$ntep1 = maxsteps - maxsteps \frac{cond1}{tol1}$$

$$ntep2 = maxsteps - maxsteps\,\frac{cond2}{tol2}$$

The formulas used for the determination of nstep1 and nstep2 are designed to avoid situations whereby cond1 and/or cond2 are very close (but not larger than) tol1 and/or tol2, respectively, indicating that no mesh refinement is needed.
In this case, if a non-adapted advancing solution is used, the solution is advanced by the a fixed number of steps ($maxsteps$) before cond1 and cond2 are re-evaluated, at which stage the solution maybe well beyond the usefulness of a remeshing remedy.
Therefore, for efficiency and accuracy reasons $nsteps$ takes into account the actual global error norms cond1 and cond2 to advance the solution in an adapted manner.

References

1. A. Rodriguez-Villa
   Etude theorique et experimentale de l'extrusion-souflage de corps creux en polymere
   PhD thesis, Ecole Nationale Superieure des Mines de Paris, France, 1997.

2. M. Bellet, J.F. Agassant and A. Rodriguez-Villa
   Finite element and automatic remeshing methods for the simulation of complex blow molded polymer components
   NUMIFORM '98, Huetink and Baaijens (eds), Balkema, Rotterdam, 1998.

3. J. Bonet
   Error estimators and enrichment procedures for finite element analysis of thin sheet large deformation processes
   I.J.N.M.E., vol. 37, 1573 – 1591, 1994.

4. J. Bonet and R.D. Wood
   Mesh enrichment procedures for the finite element analysis of thin sheet forming processes
   Num. Meth. In Ind. Processes, Chenot, Wood and Zienkiewicz (eds.), Balkema, Rotterdam, 221 – 227, 1992.

5. A. Selman, E. Hinton and W. Atamaz-sibai
   Edge effects in Mindlin-Reissner plates using adaptive mesh refinement
   Eng. Comput., vol 7, 217 – 226, 1990.

6. A. Selman and E. Hinton
   Two dimensional transient dynamic analysis with adaptive mesh refinement
   Comm. Num. Meth. Eng. vol 9, 825 – 836, 1993.

7. A. Selman, E. Hinton and N. Bicanic
   Adaptive mesh refinement for localised phenomena
   Comp. & Struct., vol. 63, 475 – 495, 1997.

8. A. Selman
   Error estimation for thin sheet metal forming processes
   Internal Report WB.99 / NIMR-0154, University of Twente, April 1999.

Error_Estimation_MFP Code User Guide

Program Error_Estimation_MFP is a Fortran Code to evaluate discretisation errors and new mesh densities for thin sheet metal forming processes.

The Code is divided into six parts

Part I Input of all necessary data and evaluation of the actual mesh size
Part II Thickness error estimation
Part III Geometric error estimation
Part IV Minimum size computation
Part V New mesh size evaluation
Part VI Remeshing conditions

These parts are now described.

Part I Input of all necessary data and evaluation of the actual mesh size.

Subroutine input

Inputs all necessary data.

VARIABLE	DESCRIPTION
nelem	Number of elements
npoin	Number of points
ndime	Number of dimensions (3)
nnode	Number of nodes (3)
ngaus	Number of Gauss points (3)
max_steps	Maximum number of steps for remeshing
gtol	Global tolerance (usually 0.05)
tol1	Used for mesh refinement criteria check (usually 15%)
tol2	Used for mesh refinement criteria check (usually 75%)
fact	Factor for minimum size evaluation
coord	Nodal coordinates – coord(ndime,npoin)
lnods	Connectivity – lnods(nnode,nelem)
thick	Thickness – thick(nelem)

Subroutine act_size

Evaluates the actual mesh size and element area.

VARIABLE	DESCRIPTION
act_siz	Actual mesh size – act_siz(nelem)
area	Element area – area(nelem)

Part II Evaluation of the **thickness** error estimation and mesh size distribution.

Subroutine nodav_thk

Computes the smooth thickness values at nodal points by a weighted nodal averaging technique.

VARIABLE	DESCRIPTION
thknp	Thickness at nodal points – thknp(npoin)
sthknp	Smooth thickness at nodal points – sthknp(npoin)
tarea	Total area connected to a given node – tarea(npoin)

Subroutine thick_error

Evaluates the error caused by thickness jumps over elements.

VARIABLE	DESCRIPTION
thkgp	Thickness at Gauss points – thkgp(ngaus)
sthkgp	Smooth thickness at Gauss points – sthkgp(ngaus)
elern	Element error norm – elern(nelem)

Subroutine thksize

Determines the thickness mesh size distribution.

VARIABLE	DESCRIPTION
thk_siz	Thickness size – thk_siz(nelem)

Part III Evaluation of the **geometric** error estimation and mesh size distribution.

Subroutine tangset

Determines the tangent set of axes.

VARIABLE	DESCRIPTION
g1	Tangent axis 1-direction – g1(ndime,nelem)
g2	Tangent axis 2-direction – g2(ndime,nelem)
g3	Tangent axis 3-direction (normal direct.) – g3(ndime,nelem)

Subroutine nodav_geo

Evaluates the smooth metric tensor by a weighted nodal averaging technique at each point.

VARIABLE	DESCRIPTION
sg1	Smooth tangent axis 1-direction – sg1(ndime,npoin)
sg2	Smooth tangent axis 2-direction – sg2(ndime,npoin)
sg	Smooth metric tensor – sg(ndime-1,ndime-1,npoin)

Subroutine geo_error

Loops over elements to evaluate the metric tensor, smooth metric tensor and Green Lagrange error tensor. This routine calls the following subroutines :

Subroutine metric_tensor

Evaluates the metric tensor and its determinant over each element.

VARIABLE	DESCRIPTION
g	Metric tensor – g(ndime-1,ndime-1)
detg	Determinant of the metric tensor – detg(nelem)

Subroutine smooth_metric_tensor

Evaluates the smooth metric tensor over each element as an average of the metric tensor at element nodes.

VARIABLE	DESCRIPTION
gs	Smooth metric tensor – gs(ndime-1,ndime-1)
detgs	Determinant of the smooth metric tensor – detgs(nelem)
sge	Smooth metric tensor over an element – sge(ndime-1,ndime-1)

Subroutine green_lagrange_tensor

Evaluates the Green Lagrange error tensor, contravariant metric tensor, contravariant Green Lagrange tensor and determines the geometric error norm.

VARIABLE	DESCRIPTION
e	Green Lagrange error tensor – e(ndime-1,ndime-1)
e33	Deformation in the normal direction
gc	Contravariant metric tensor – gc(ndime-1,ndime-1)
ec	Contravariant Green Lagrange error tensor – ec(ndime-1,ndime-1)
gelern	Geometric error norm – gelern(nelem)

Subroutine geosize

Determines the geometric mesh size distribution.

VARIABLE	DESCRIPTION
geo_siz	Geometric size – geo_siz(nelem)

Part IV Evaluation of the minimum size allowed

Subroutine minsize

Determines the minimum mesh size distribution.

VARIABLE	DESCRIPTION
min_siz	Minimum size allowed – min_siz(nelem)

Part V Evaluation of the new mesh size

Subroutine newsize

Determines the new mesh size distribution.

VARIABLE	DESCRIPTION
new_siz	New mesh size – new_siz(nelem)

Part VI Checking if mesh refinement is needed

Subroutine criteria

Checks the criteria for mesh refinement.

VARIABLE	DESCRIPTION
meshind	Mesh refinement indicator
cond1	Mean value of the refinement condition
cond2	Maximum value of the refinement condition
nsteps	Adaptive number of steps to advance the solution

See notes on error estimation for thin sheet metal forming processes.

Input Data Format

nelem,npoin,ndime(3)**,nnode**(3),**ngaus**(3**) , maxsteps,gtol**(5%),**tol1**(15%),
tol2(75%), **fact**(2-4)
do ipoin = 1,npoin
read(unit,format) **jpoin,(coord(idime,ipoin),idime=1,ndime)**
enddo
do ielem= 1,nelem
read(unit,format) **jelem,(lnods(inode,ielem),inode=1,nnode),thick(ielem)**
enddo

Benchmark tests

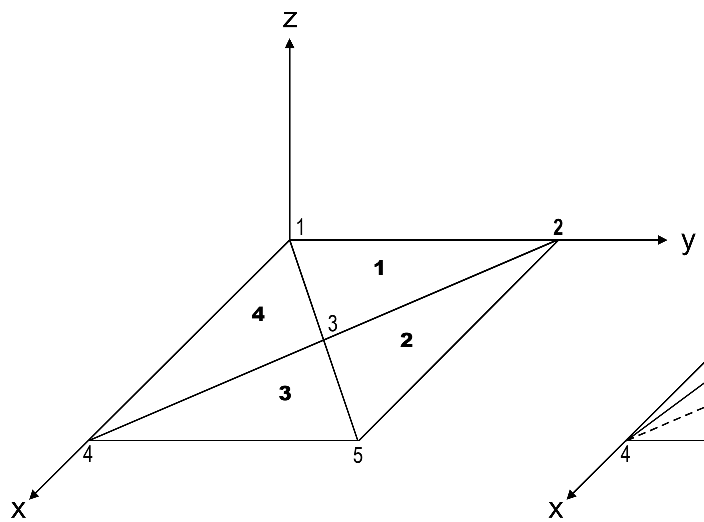

Figure 1.

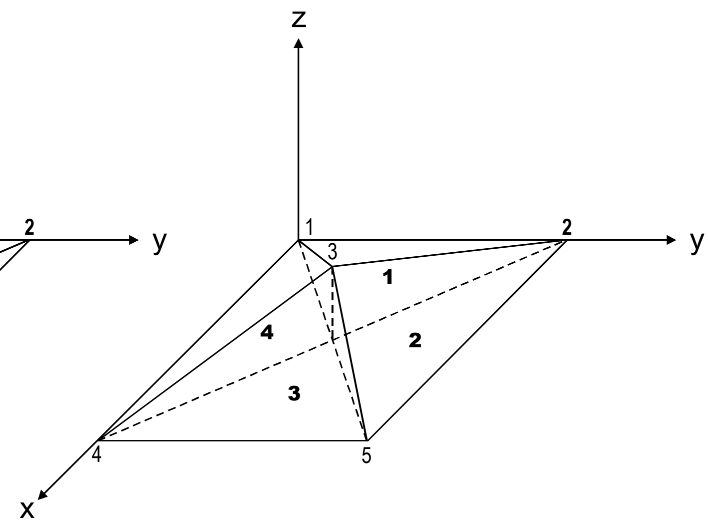

Figure 2.

Test number 1

In the Error_Estimation_MFP data file provided, a flat sheet with a constant thickness of 0.1 is considered in the xy plane (Figure 1). In this case, the thickness and geometric errors for all elements are nil and, consequently, the thickness and geometric sizes are set to the maximum size.

Test number 2

Consider a flat surface and set a thickness of 0.1 for elements 1 and 3 and a thickness of 0.2 for elements 2 and 4. In this case, note that while the geometric error remains nil, the thickness error is increasing leading to smaller thickness (new) sizes.

Test number 3

Set a constant thickness of 0.1 for all elements, and start moving node 3 in the z-direction (Figure 2). Note that as node 3 is moved away from the initial sheet plane, the geometric errors increase for all elements.

Test number 4

Position node 3 at (5,5,5) and set a thickness of 0.1 for elements 1 and 3 and a thickness of 0.2 for elements 2 and 4. Note that now both the thickness and geometric errors have risen leading to smaller new mesh sizes.

Test number 5

Position node 3 at (7,5,5) and repeat test 4. Note that symmetric values are obtained for element 2 and 4 and that a smaller new mesh size is obtained for element 3 than for element 1.