MD-Vol. 75/AMD-Vol. 219, Advances in Failure Mechanisms in Brittle Materials
ASME 1996

COMPUTATIONAL MODELING OF GEOMETRIC AND MATERIAL NONLINEARITIES WITH AN APPLICATION TO IMPACT DAMAGE IN BRITTLE MATERIALS

Horacio D. Espinosa, Gordon Emore, Pablo Zavattieri

School of Aeronautics and Astronautics
Purdue University
West Lafayette, IN 47907-1282

ABSTRACT

A model is presented for the dynamic finite element analysis of large-strain, high strain rate deformation behavior of materials. Both an isochoric plastic model for metals and a multiplane microcracking model for ceramics are introduced, and a temperature-dependent model is included. In addition, algorithms are presented for correcting finite element mesh distortion through mesh rezoning, optimization, and refinement. A surface-defined multibody contact algorithm designed to handle large relative displacements between bodies is included, with additions for friction and cohesive interfaces. A test example is examined, simulating a modified Taylor rod impact test in which an aluminum anvil strikes a ceramic specimen. Velocities are recorded at the free end of the ceramic and compared to experimental results. It is observed that the simulation produces the same overall features as the experimental data.

INTRODUCTION

There is a class of problems in which material and geometric nonlinearities play a major role in understanding the physics of the problems. This group includes problems in high speed machining, metal forming, and impact-induced damage in ductile and brittle materials. What make these problems special is the fact that the deformations involved are both severe and takes place very quickly, resulting in large strains and strain rates, high temperatures and stresses, and unusual friction conditions. In machining, for example, the surfaces of workpieces have been estimated to have plastic strains as high as 500 percent, as seen in Komanduri, 1993. Experimental observations of metal cutting operations show that deformations occur in concentrated regions, in which material is subject to shear deformation at very high strain rates ($10^3 - 10^5$), Oxley, 1989. A similar scenario is observed in impact and ballistic penetration, Zukas, 1990.

Experimental investigations show that material properties, such as stress-strain properties, are strongly affected by strain rate and temperature during plastic deformation, Duffy, 1979, Clifton and Klopp, 1985, Nicholas & Rajendran, 1990. For the same strain, the corresponding stress increases with strain rate due to viscous effects. In addition, heat generated by plastic processes and friction increases the temperature of the material, leading to thermal softening which can result in material instabilities. Temperatures in excess to $600°C$ have been observed in machining experiments performed on 1020 steel, Shih & Yang, 1993. High temperatures resulting from plastic deformation were also measured within shear bands in maragin steel and Ti alloys, Zhou et al., 1996.

In general, physical experiments do not always provide direct information on material behavior and failure. Therefore, numerical simulations are useful in helping researchers to obtain insight into the physics of deformation and fracture of advanced materials. However, from a computational standpoint, the class of problems previously mentioned present several difficulties. Large relative deformations are occurring very quickly in well defined moving domains. Strain and temperature gradients can be extremely sharp requiring a high degree of spatial and temporal resolution of the field variables.

Most FEM calculations for solid mechanics use a Lagrangian formulation, Belytschko, 1983, Benson, 1992. This means that as the mesh becomes deformed, errors may accumulate from highly distorted elements. A greatly deformed element may become unsuitable for further updates in the calculation, e.g., Yang et al., 1989, Batra and Ko, 1992, and Ortiz and Quigley, 1991. In addition, as elements are distorted, the time step required for explicit integration algorithms (which are only conditionally stable) tends to dwindle such that completing a dynamic calculation may become extremely expensive computationally, Espinosa et al., 1995. The initial mesh may also become inappropriate as time progresses, because high strain gradients may develop in areas of the mesh with insufficient resolution to accurately model geometric and material nonlinearities.

There are several methods for dealing with these problems. Highly distorted meshes may be fixed by moving and *optimizing* node locations, as in Zienkiewicz & Taylor, 1993b. Nodes may also be moved closer to regions of interest using *adaptive meshing* algorithms. Several contributions have been made in this direction, e.g., Zhu et al., 1991, and Batra & Hwang, 1993. Alternatively, a mesh may be replaced completely with a brand new mesh in an operation called *remeshing* or *rezoning.* Additional nodes may be added in regions of interest to produce more elements in an *h-refinement*, thereby creating more resolution where large temperature and displacement gradients are occurring, Marusich and Ortiz, 1995. Another way of dealing with mesh distortion is by using arbitrary Lagrangian-Eulerian (ALE) methods, Donea, 1983, Liu et al., 1988, Benson, 1992, Dvorkin and Petocz, 1992. As featured in Stillman et al., 1993, the ALE method is able to continuously rebuilds the boundaries in an Eulerian fashion, as part of the regular smoothing operation, resulting in good accuracy and reducing CPU time with respect to pure Lagrangian formulations.

The development of new and faster computers with parallel capabilities, as well as the manufacturing of ultra fast speed cameras, detectors, and oscilloscopes, open new frontiers for the investigation of the mechanics and material issues discussed above. As an example of how experiments and numerical simulations are combined to examine material inelasticity and failure, we here present a finite element model to simulate dynamic contact, and geometric and material nonlinearities resulting from high speed impact. The model includes a temperature and strain-rate dependent material model for large deformation plasticity, microcracking for the case of brittle materials, and a multi-body contact law with a variable-coefficient friction law and heat conduction. In addition, the capability to rezone, refine, optimize, and update the finite element mesh is included to allow the calculation to continue in a timely fashion through extremely severe deformations. The features of the developed computational algorithms are illustrated by examining the high velocity impact of a T-shaped aluminum rod against a ceramic rod, Wise and Grady, 1994. Interferometrically measured velocities are used to assess the quality of the proposed models.

THE STRESS AND DEFORMATION MODEL

Field Equations

In our formulation, the field equations describing the material response of a body use a Lagrangian approach. Consider a solid with volume B_0 in the reference configuration, and a deformation process characterized by the mapping $\mathbf{x}(\mathbf{X}, t)$. Then a material point initially at $\mathbf{X}$ will be located at $\mathbf{x} = \mathbf{X} + \mathbf{u}$ after deformation, in which $\mathbf{u}$ is the displacement vector, as shown in Figure 1. A displacement based finite element formulation is obtained from the weak form of the momentum balance or dynamic principle of virtual work. At time t the weak form is given by

$$\int_{B_o} [\nabla_o \mathbf{T}^o + \rho_o(\mathbf{b}_o - \mathbf{a})] \cdot \eta dB_o = 0 \tag{1}$$

$$\int_{B_o} \mathbf{T}^o : \nabla_o \eta dB_o - \int_{B_o} \rho_o(\mathbf{b}_o - \mathbf{a}) \cdot \eta dB_o - \int_{S_{o\sigma}} \mathbf{t} \cdot \eta dS_o = 0 \tag{2}$$

where $\mathbf{T}^0$ is the first Piola-Kirchhoff stress tensor at time t; $\mathbf{b}_o$, $\mathbf{a}$, and $\mathbf{t}$ are the body force vector, acceleration vector, and boundary traction vector on volume B_o and boundary $S_{o\sigma}$, respectively. Virtual displacement field η is assumed to be admissible, and ρ_o represents the material density in reference configuration. The symbol ∇_0 denotes the material gradient with respect to the reference configuration. and ':' is used to denote

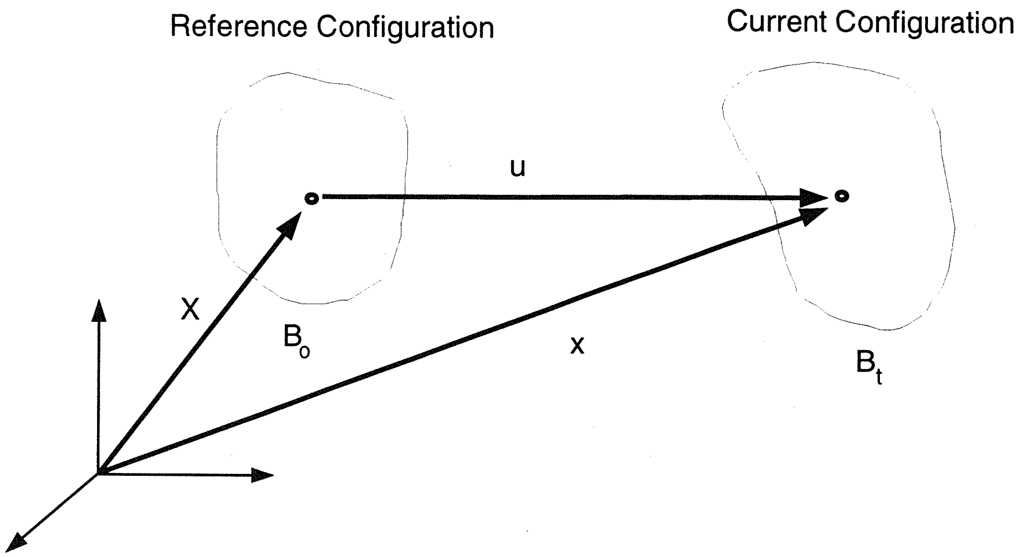

Figure 1: **The reference and current configurations.**

the inner product between second order tensors, e.g., $\mathbf{A} : \mathbf{B} \equiv A_{ij}B_{ji}$, where the summation convention on repeated indices is implied. Since the Kirchhoff stress is given by $\tau = \mathbf{F}\mathbf{T}^o$, wherein $\mathbf{F}$ is the deformation gradient $\nabla_o\mathbf{x}$ at time t, another form of the weak form of the momentum balance, in terms of spatial quantities, is given by

$$\int_{B_o} \tau : \nabla^s \eta dB_o - \int_{B_o} \rho_o(\mathbf{b}_o - \mathbf{a}) \cdot \eta dB_o - \int_{S_{o\sigma}} \mathbf{t} \cdot \eta dS_o = 0. \qquad (3)$$

in which superscript ^{s} stands for the symmetric part of the tensor and ∇ is the spatial deformation tensor. As equation 3 shows, the reduced weak form states that the work done by the stresses τ over volume B_o equals the work done by applied body forces, inertia forces, and surface tractions.

Large Deformation Plasticity Model

The plasticity constitutive law used here is based on an atomic lattice model for metals to describe elastic and plastic material behavior. Consider a crystal lattice being deformed from its initial undeformed configuration B_o to its current configuration B_t at time t. Locally, the deformation of the crystal is fully defined by the deformation gradient field $\mathbf{F}$. The total deformation $\mathbf{F}$ is the result of two main mechanisms of deformation: dislocation motion within the active slip systems of the crystal and lattice distortion. Following Lee, 1969, this points to a multiplicative decomposition

$$\mathbf{F} = \mathbf{F}^e\mathbf{F}^p \qquad (4)$$

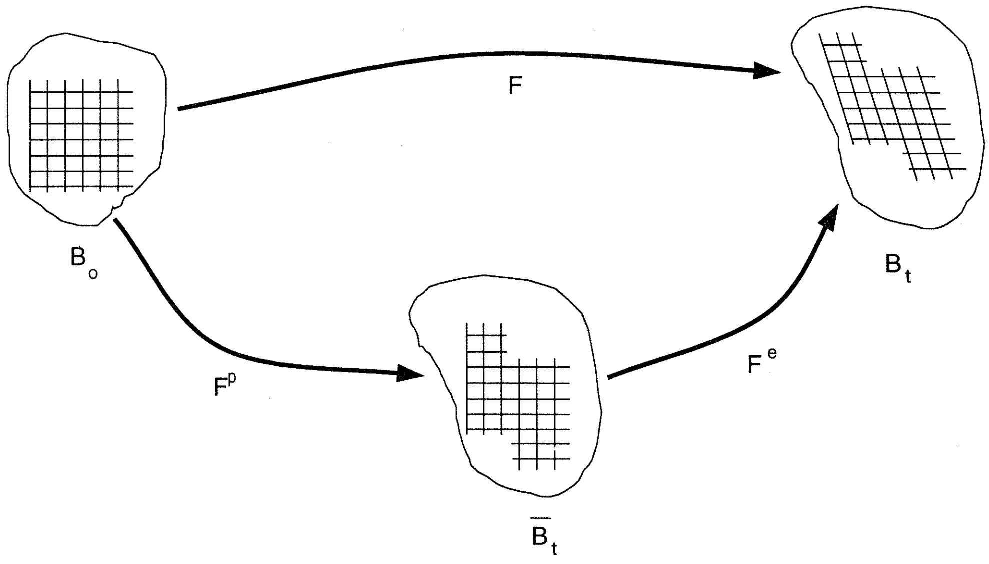

Figure 2: **Deformation gradient decomposition.**

of the deformation gradient $\mathbf{F}$ into a plastic part $\mathbf{F}^p$, defined as the cumulative effect of dislocation motion, and an elastic part $\mathbf{F}^e$, which describes the distortion of the lattice. Following Teodosiu, 1970, Rice, 1971, Mandel, 1972, Hill & Rice, 1972, Havner, 1973, and Asaro & Rice, 1977, we shall assume that $\mathbf{F}^p$ leaves the crystal not only essentially undistorted, but also unrotated, as seen in Figure 2. Thus, the rotation of the lattice is contained in $\mathbf{F}^e$. This choice of kinematics uniquely determines the decomposition 4. The plastic part $\mathbf{F}^p$ of the deformation gradient defines a collection of plastically deformed local configurations which are collectively referred to as the intermediate configuration $\overline{B}_t$.

Insight into the structure of the constitutive relations can be derived from work conjugacy considerations. By virtue of the decomposition 4, the deformation power per unit undeformed volume takes the form

$$\mathbf{T}^o : \dot{\mathbf{F}} = \overline{\mathbf{T}}^o : \dot{\mathbf{F}}^e + \overline{\mathbf{\Sigma}} : \overline{\mathbf{L}}^p \tag{5}$$

where $\mathbf{T}^o$ is the first Piola-Kirchhoff stress tensor, and

$$\overline{\mathbf{T}}^o = \mathbf{T}^o \mathbf{F}^{pT} \tag{6}$$

$$\overline{\mathbf{\Sigma}} = \mathbf{F}^{eT} \mathbf{T}^o \mathbf{F}^{pT} \tag{7}$$

$$\overline{\mathbf{L}}^p = \dot{\mathbf{F}}^p \mathbf{F}^{p-1} \tag{8}$$

Here, $\overline{\mathbf{T}}^o$ defines a first Piola-Kirchhoff stress tensor relative to the intermediate configuration $\overline{B}_t$, and $\overline{\Sigma}$ is a stress measure conjugate to the plastic velocity gradients $\overline{\mathbf{L}}^p$ on $\overline{B}_t$, Mandel, 1972.

The work conjugacy relations expressed in (9) suggest plastic flow rules and elastic stress-strain relations of the general form, Mohan, 1991,

$$\begin{aligned} \overline{\mathbf{L}}^p &= \overline{\mathbf{L}}^p(\overline{\Sigma}, \overline{\mathbf{Q}}) & (9) \\ \overline{\mathbf{T}}^o &= \overline{\mathbf{T}}^o(\mathbf{F}^e) & (10) \end{aligned}$$

Here, $\overline{\mathbf{Q}}$ denotes some suitable set of internal variables defined on the intermediate configuration, for which suitable equations of evolution, or hardening laws, are to be supplied. It should be noted that 9 determines uniquely both the plastic deformation and the rate of plastic rotation. Because, by the choice of kinematics, $\mathbf{F}^p$ is invariant with respect to rigid body motions superimposed on the current configuration, 9 is automatically material frame indifferent. By contrast, objectivity imposes non-trivial restrictions on the form of 10. It can be shown, Mohan, 1991, that the most general form of 10 consistent with the principle of material frame indifference is

$$\begin{aligned} \overline{\mathbf{T}}^o &= \mathbf{F}^e\overline{\mathbf{S}}(\overline{\mathbf{C}}^e), & (11) \\ \overline{\mathbf{C}}^e &= \sqrt{\mathbf{F}^{eT}\mathbf{F}^e} & (12) \end{aligned}$$

where $\overline{\mathbf{S}} = L(\frac{1}{2}\ln\overline{\mathbf{C}}^e) = L\overline{\mathbf{E}}^e$ is the symmetric second order Piola-Kirhhoff stress tensor relative to the intermediate configuration $\overline{B}_t$, and $\overline{\mathbf{C}}^e = \sqrt{\mathbf{F}^e\mathbf{F}^{eT}}$ is the elastic right Cauchy-Green deformation tensor. In addition, $\overline{\mathbf{E}}^e$ is the logarithmic strain measure and L the material stiffness tensor. For an elastic isotropic solid L is given by

$$L = 2\mu\mathbf{I} + (k - \frac{2}{3}\mu)\mathbf{1}\otimes\mathbf{1}, \quad (13)$$

in which μ is the shear modulus, k is the bulk modulus, $\mathbf{I}$ is the identity tensor, and $\mathbf{1}$ is the unity vector.

In summary, the elastic and plastic portions of the deformation are separated by the multiplicative decomposition $\mathbf{F} = \mathbf{F}^e\mathbf{F}^p$ as outlined above, subjected to the restrictions $\det\mathbf{F}^e > 0$ and $\det\mathbf{F}^p = 1$. Moreover, from 8 it can be seen that the evolution equation for $\mathbf{F}^p$ follows the flow rule

$$\begin{aligned} \dot{\mathbf{F}}^p &= \overline{\mathbf{L}}^p\mathbf{F}^p; & (14) \\ \overline{\mathbf{L}}^p &= \mathbf{W}^p + \mathbf{D}^p, & (15) \end{aligned}$$

wherein $\overline{\mathbf{L}}^p$ is the plastic part of the velocity gradient $\overline{\mathbf{L}}$. The above assumes that $\mathbf{W}^p \equiv skw\overline{\mathbf{L}}^p = \mathbf{0}$ and that $\mathbf{D}^p = sym\overline{\mathbf{L}}^p = \mathbf{D}^p(\overline{\mathbf{S}}, \overline{\mathbf{Q}}) = \dot{\epsilon}^p\overline{\mathbf{N}}$. Here, $\overline{\mathbf{N}}$ is the plastic flow direction and $\dot{\epsilon}^p$ is the effective plastic strain rate. The plastic flow direction is

Constitutive	$\Longleftrightarrow$	*Incremental*
$\mathbf{F} = \mathbf{F}^e\mathbf{F}^p$	$\Longleftrightarrow$	$\mathbf{F}_{n+1} = \mathbf{F}^e_{n+1}\mathbf{F}^p_{n+1}$
$\dot{\mathbf{F}}^p\mathbf{F}^{p-1} = \dot{\epsilon}^p\overline{\mathbf{N}}(\overline{\mathbf{S}},\overline{\mathbf{Q}})$	$\Longleftrightarrow$	$\mathbf{F}^p_{n+1} = \exp(\Delta\epsilon^p\overline{\mathbf{N}}_{n+1})\mathbf{F}^p_n$
$\overline{\mathbf{S}} = L(\frac{1}{2}\ln\mathbf{C}^e)$	$\Longleftrightarrow$	$\overline{\mathbf{S}}_{n+1} = L(\frac{1}{2}\ln\mathbf{C}^e_{n+1})$
$\dot{\overline{\mathbf{Q}}} = \dot{\epsilon}^p\overline{\mathbf{H}}(\overline{\mathbf{S}},\overline{\mathbf{Q}})$	$\Longleftrightarrow$	$\overline{\mathbf{Q}}_{n+1} = \overline{\mathbf{Q}}_n + \Delta\epsilon^p\overline{\mathbf{H}}_{n+1}$

Table 1: **Numerical integration algorithm**

given as a function of the deviatoric stress tensor $\overline{S}_{dev}$, and the effective stress defined by

$$\overline{\mathrm{N}} = \frac{3}{2}\frac{\overline{\mathrm{S}}_{dev}}{\overline{\sigma}} \tag{16}$$

$$\overline{\sigma} = \sqrt{\frac{3}{2}\overline{\mathrm{S}}_{dev} : \overline{\mathrm{S}}_{dev}}. \tag{17}$$

The effective plastic strain rate, $\dot{\epsilon}^p$, is a function of the effective stress $\overline{\sigma}$, temperature T, and the internal variables $\overline{\mathbf{Q}}$, viz.,

$$\dot{\epsilon}^p = f(\overline{\sigma}, \overline{\mathbf{Q}}, T). \tag{18}$$

For a viscoplastic representation of the J_2 -flow theory, we have

$$\dot{\epsilon}^p = \dot{\epsilon}^p_o[(\frac{\overline{\sigma}}{\mathrm{s}(\epsilon^p, T)})^\alpha - 1], \qquad \overline{\sigma} > \mathrm{s} \tag{19}$$

$$\dot{\epsilon}^p = 0 \qquad \overline{\sigma} \leq \mathrm{s} \tag{20}$$

$$f(\overline{\sigma}, \overline{\mathbf{Q}}, T) = \dot{\epsilon}^p_o[(\frac{\overline{\sigma}}{\mathrm{s}(\epsilon^p, T)})^\alpha - 1] \tag{21}$$

$$\mathrm{s}(\epsilon^p, T) = \sigma_o(1 + \frac{\epsilon^p}{\epsilon^p_o})^\beta[1 - (\frac{T - T_o}{T_m - T_o})^\gamma] \tag{22}$$

such that ϵ^p is the accumulated plastic strain, σ_o is the initial material strength, T_m is the melting temperature of the material, and α, β,and γ are the rate sensitivity, hardening, and thermal softening exponents. The material strength $\mathbf{s}$ and ϵ^p are internal variables in $\overline{\mathbf{Q}}$. When translated into incremental formuli, the constitutive model transforms the elastic part of the deformation gradient into new stresses $\overline{\mathbf{S}}$, as shown in Table 1. In this model $\overline{\mathbf{H}}$ represents the function defining the evolution of internal variables $\overline{\mathbf{Q}}$. The exponential law for $\mathbf{F}^p$ at time t_{n+1} is derived from

Multiplane Microcracking Model

In this section the inelastic response of ceramics is modeled through a microcracking multiple-plane model based on a dilute approximation (Taylor model). Our formulation overlaps with some theories in which multiple-plane representations of inelasticity are derived, e.g., Seaman & Dein, 1983, Bazant & Gambarova, 1984, and Ju & Lee, 1991.

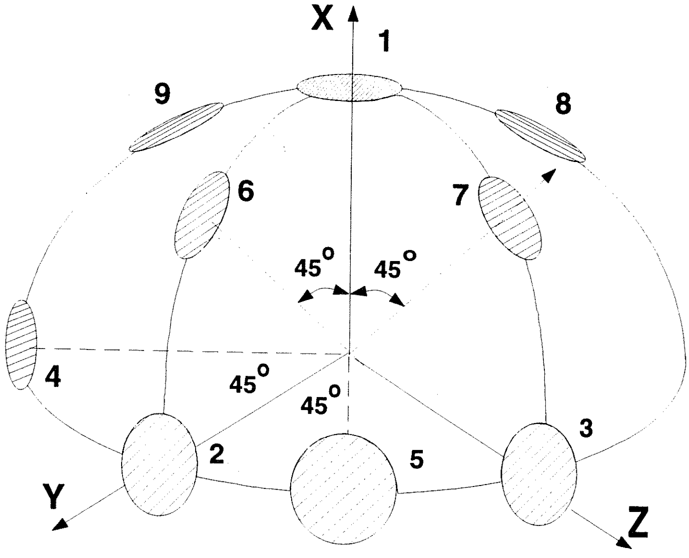

Figure 3: **Schematic of microcracking multiple-plane model.**

The basic assumption in the microcracking multiple-plane model is that microcracking and/or slip can occur on a discrete number of orientations (see Figure 3). Slip plane properties (friction, initial size, density, etc.) and their evolution are *independently* computed on each plane. The macroscopic response of the material is based on an additive decomposition of the strain tensor into an elastic part and an inelastic contribution arising from the presence of microcracks within the solid. In contrast to scalar representations of damage, e.g., Rajendran, 1992, the formulation by Espinosa, 1989 is broad enough to allow the examination of *damage induced anisotropy* and damage localization in the interpretation of impact experiments.

For a representative volume B of an elastic solid containing penny-shaped microcracks with a density $N^{(k)}$, the average inelastic strains are given by

$$\epsilon_{ij}^{c} = \sum_{k=1}^{9} N^{(k)} S^{(k)} \frac{1}{2} (\bar{b}_i^{(k)} n_j^{(k)} + n_i^{(k)} \bar{b}_j^{(k)}) \qquad (23)$$

where the subindex k is used to label the orientations, $S^{(k)}$ denotes the surface of a microcrack on orientation k, $n^{(k)}$ the corresponding unit normal, and $\bar{b}^{(k)}$ the average displacement jump vector across $S^{(k)}$.

If the resolved normal traction acting on the microcracks on orientation k is tensile, the average displacement jump vector resulting from an applied stress field σ is given by

$$\bar{b}_i^{(k)} = \frac{1}{S^{(k)}} \int_{S^{(k)}} b_i^{(k)} dS = \frac{16(1-\nu^2)}{3E(2-\nu)} a^{(k)} (2\sigma_{ij} n_j^{(k)} - \nu \sigma_{jl} n_j^{(k)} n_l^{(k)} n_i^{(k)}), \tag{24}$$

in which E and ν are the Young's modulus and Poisson's ratio of the uncracked solid, and a^k is the radius of the penny-shaped microcracks on orientation k. By contrast, if the normal traction is compressive, the microcracks are closed and the average displacement jump is given by

$$\bar{b}_i^k = \frac{32(1-\nu^2)}{3\pi E(2-\nu)} a^k f_i^k \tag{25}$$

where $\mathbf{f}^k$ is the effective shear traction vector on orientation k given by

$$f_i^k = (\tau^k + \mu \sigma_n^k)(n_\tau)_i^k, \tag{26}$$

in 26, μ is the friction coefficient of the microcrack faces, τ^k and σ_n^k are the resolved shear stress and the normal stress acting on microcracks with orientation k, respectively, and n_τ^k is the unit vector in the direction of the resolved shear traction. Embodied in 25 is the notion that f^k provides the effective driving force for the sliding of the microcracks.

In order to compute the inelastic strain tensor at all times, it becomes necessary to follow the evolution of the microcrack radius a^k in the selected orientations. Following Freund, 1990, an equation of evolution for a in the case of mixed mode loading can be derived, viz.,

$$\dot{a}^k = m^{\pm} c_R [1 - (K_{IC}/K_{eff}^k)^{n^{\pm}}] \geq 0, \tag{27}$$

in which n$^{\pm}$ and m$^{\pm}$ are phenomenological material constants which may have different values in tension and compression, c_R is the Rayleigh wave speed, K_{IC} is the material toughness, and K_{eff}^k is an effective stress intensity factor. For mixed mode conditions, K_{eff}^k is derived by considering an average energy release rate associated with an increase in radius of the microcracks, namely,

$$\mathcal{G}^k = \frac{1}{2\pi} \int_0^{2\pi} \frac{1-\nu^2}{E} \left[K_I^2 + K_{II}^2 + K_{III}^2/(1-\nu) \right] d\theta \tag{28}$$

from which the following expression for K_{eff}^k is obtained,

$$K_{eff}^k = \sqrt{\frac{\mathcal{G}^k E}{1-\nu^2}}. \tag{29}$$

The general structure of these constitutive equations corresponds to that of a solid with a damage-induced anisotropic stress-strain relation with elastic degradation. In particular, the effective behavior of the solid is predicted to be rate dependent due to crack kinetics effects. From a computational standpoint, this ensures numerical reliability and mesh independence, according to Needleman,1988, and Espinosa,1989. This is in contrast to quasi-static formulations of damage for which the governing equations become ill-posed in the softening regime, as in Sandler & Wright, 1984.

If the material is subjected to a predominantly tensile stress state, microcracks along orientations perpendicular to the direction of maximum tensile stresses will grow according to Eq. (5). In this case, significant dilation is expected due to mode I crack opening. If a predominantly compressive state of stress with shear is imposed, then crack opening is inhibited but inelasticity is manifested by the growth of penny-shaped cracks in modes II and III (shear modes).

Finite Element Discretization

Discretization of equation 3 defines a system of nonlinear ordinary differential equations which can be solved for the updated deformation $\mathbf{x}_{n+1}$. A displacement based finite element formulation is obtained by expressing field variables at any point in an element as a function of nodal quantities and the element shape functions in the reference configuration:

$$
\begin{aligned}
u_{ia} &= \sum_{a=1}^{NEN} N_a(\xi) u_i^a && (30)\\
v_{ia} &= \sum_{a=1}^{NEN} N_a(\xi) v_i^a && (31)\\
a_{ia} &= \sum_{a=1}^{NEN} N_a(\xi) a_i^a && (32)
\end{aligned}
$$

wherein u_{ia}, v_{ia}, and a_{ia} are the displacement, velocity, and acceleration at the point of interest. The sum is taken over the number of nodes in the element, NEN, while $N_a(\xi)$ are the shape functions relating nodal quantities u_i^a, v_i^a, and a_i^a to the point of interest. The vector ξ contains the natural coordinates of the point of interest in the isoparametric master element. In 2-D calculations, ξ expands to (ξ, η), where the components are functions of the area coordinates (ξ_1, ξ_2, ξ_3) for a triangular element. Substitution of the discretized variables into 3 leads to the following system of equations,

$$
\begin{aligned}
M_{iab} a_i^b &= f_{ia}^{ext} - f_{ia}^{int} && (33)\\
f_{ia}^{int} &= \sum_e \int_{B_o^e} n_{a,j} \tau_{ij} dB_o = \sum_e \int_{B_o^e} \mathbf{B}^T \tau dB_o && (34)\\
f_{ia}^{ext} &= \sum_e \int_{B_o^e} \rho_o b_{oi} N_a^e dB_o + \int_{\partial S_o^e} \bar{t}_i N_a dS_o && (35)\\
M_{iab} &= \int_{B_o^e} \rho_o N_b^e N_a^e dB_o, && (36)
\end{aligned}
$$

in which f_{ia}^{int}, f^{ext}, and M_{iab} are the internal nodal forces, the external nodal forces, and the lumped mass matrix, respectively. The integrations of the body forces, inertia forces, and stress contributions take place over an element volume B_o^e, while the applied tractions are integrated over element surface ∂S_o^e. Dynamic equilibrium requires that internal and external forces are balanced by the inertia forces. The shape function derivatives $N_{a,I}$ and $n_{a,j}$ are defined in the reference and current configurations. The body forces and surface tractions are applied upon the volume B_o, and surface ∂S_o^e. In the expression for f^{int}, the Kirchoff stress can be expressed in terms of the second Piola-Kirchhoff stress tensor $\overline{\mathbf{S}}$ in the intermediate configuration, and the elastic part of the deformation gradient, $\mathbf{F}^e$, namely,

$$\tau = \mathbf{F}^e \overline{\mathbf{S}} \mathbf{F}^{eT}. \tag{37}$$

Direct Time Integration Algorithm

An explicit central-difference integration algorithm is being used to integrate in time. This algorithm is presented in Table 2. As a central difference method, initial displacements and velocities $\mathbf{u}^o$ and $\mathbf{v}^o$ are required, while initial accelerations $\mathbf{a}^0$ may be calculated from initial applied tractions $\mathbf{f}^{ext^0}$, and initial internal forces $\mathbf{f}^{int^0}$.

At each time step n, the nodal accelerations must first be corrected for any time-dependent changes in the traction boundary conditions. Then the displacements at time $n+1$ are computed using the corrected acceleration and the displacements and velocities at time step n. The stresses are updated next using the material constitutive law, followed by computation of internal forces as a function of the Kirchhoff stress tensor. Finally, acceleration and velocity at time $n+1$ are computed before incrementing the time.

This explicit integration method is very useful for studies in which high rates of loading are expected. The time steps used by these explicit calculations are limited by stability, so care must be taken in finite deformation dynamic calculations to ensure that waves do not propagate through the mesh faster than the material wave speeds. To this end, the time step is calculated dynamically from the maximum element frequency in the mesh ω_{max}, such that $\Delta t \leq 2/\omega_{max}$ Flanagan & Belytschko, 1984, derived the following estimate of ω_{max} for an N-noded isoparametric element,

$$\omega_{max}^2 \leq N \left(\frac{\hat{\lambda} + 2\hat{\mu}}{\rho} \right) \frac{B_{iI} B_{iI}}{A^2} \tag{38}$$

in which $B_{iI}B_{iI}$ is the trace of $[B][B]^T$, and the area A_o is found as $C_{IJ}X_IY_J$, where

$$C_{IJ} = \int\int_A \left(\frac{\partial N_I}{\partial \xi} \frac{\partial N_J}{\partial \eta} - \frac{\partial N_I}{\partial \eta} \frac{\partial N_J}{\partial \xi} \right) d\eta d\xi, \tag{39}$$

and N_I are the element shape functions. For the case of a 6-noded isoparametric trian-

1. Initial conditions:

$$n = 0$$
$$\mathbf{u}^0 = \overline{\mathbf{u}}^0$$
$$\mathbf{v}^0 = \overline{\mathbf{v}}^0$$
$$\mathbf{a}^0 = \mathbf{M}^{-1}(\mathbf{f}^{ext^0} - \mathbf{f}^{int^0}).$$

2. Correct accelerations due to changes in boundary conditions:

$$\hat{\mathbf{a}}^n = \mathbf{M}^{-1}(\mathbf{f}^{ext^{n+1}} - \mathbf{f}^{ext^n})$$

3. Update displacements:

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \triangle t \mathbf{v}^n + \frac{1}{2}\triangle t^2 \hat{\mathbf{a}}^n.$$

4. Update the second Piola Kirchhoff stress tensor $\overline{\mathbf{S}}$ at each element (see Table 1)

5. Compute internal force vector:

$$\mathbf{f}^{int} = \int_{B_o} \mathbf{B}^T \tau dB_o; \quad \tau = \mathbf{F}^e \overline{\mathbf{S}} \mathbf{F}^{eT}.$$

6. Solve for accerations:

$$\mathbf{a}^{n+1} = \mathbf{M}^{-1}(\mathbf{f}^{ext^{n+1}} - \mathbf{f}^{int^{n+1}}).$$

7. Update velocity vector:

$$\mathbf{v}^{n+1} = \mathbf{v}^n + \frac{\Delta t}{2}(\hat{\mathbf{a}}^n + \mathbf{a}^{n+1}).$$

8. $n = n + 1$, if $n < n_{max}$ go to step (ii), else stop.

Table 2: **Explicit integration algorithm**

gular element, C_{IJ} is computed to be the 6x6 matrix

$$C_{IJ} = \frac{1}{6}\begin{pmatrix} 0 & 3 & 7 & 10 & -10 & -10 \\ -3 & 0 & -7 & -10 & 10 & 10 \\ -7 & 7 & 0 & 0 & -8 & 8 \\ -10 & 10 & 0 & 0 & -8 & 8 \\ 10 & -10 & 8 & 8 & 0 & -16 \\ 10 & -10 & -8 & -8 & 16 & 0 \end{pmatrix}. \tag{40}$$

MECHANICAL CONTACT

Contact algorithms allow the study of interactions between bodies without the need for applying explicit pressure distributions on the surface being studied. This algorithm is based on the 2-D dynamic model presented in PRONTO, from Taylor & Flanagan, 1987, wherein the method considers contact to be a kinematic constraint. As such, the algorithm must alter the accelerations of the surface nodes so that the surfaces do not interpenetrate. In addition, a velocity-dependent friction model is included. This form of contact was used because it operates on surfaces, not contact elements, as is usual. This surface-based contact algorithm allows contact between bodies that undergo large relative displacements as they move. It also allows the easy incorporation of a velocity-dependent friction model where the friction coefficient is made a function of pressure and temperature. In this section we start with a description of the contact algorithm as presented in Taylor & Flanagan, 1987, followed by an extension to model cohesive interfaces.

Surface Definition

The contact algorithm must keep track of the relative positions of the two surfaces. This requires that it store an array listing the surface nodes for each body. In our FEM code, which is a modified version of FEAP, Zienkiewicz and Taylor, 1993a, the mesh is generated in blocks, and a subroutine is called to generate the needed arrays of surface nodes.

To begin, the subroutine identifies all corner nodes on the surface of the block. This is currently done in a loop over all of the elements, wherein each corner node accumulates the difference between the node numbers of the other two corner nodes. The end result of this is that all interior corner nodes accumulate a sum of zero, while the boundary corner nodes have a nonzero sum. Then, starting with the first boundary corner node, the algorithm examines the elements attached to the node to find the next corner node on the boundary, in a counter-clockwise direction. This is easily enough done as the element connectivities are also defined in a counter-clockwise direction. As each new boundary corner node is added to the surface array, the midnode between it and the previous node is also added. The algorithm repeats the process until it returns to the starting point.

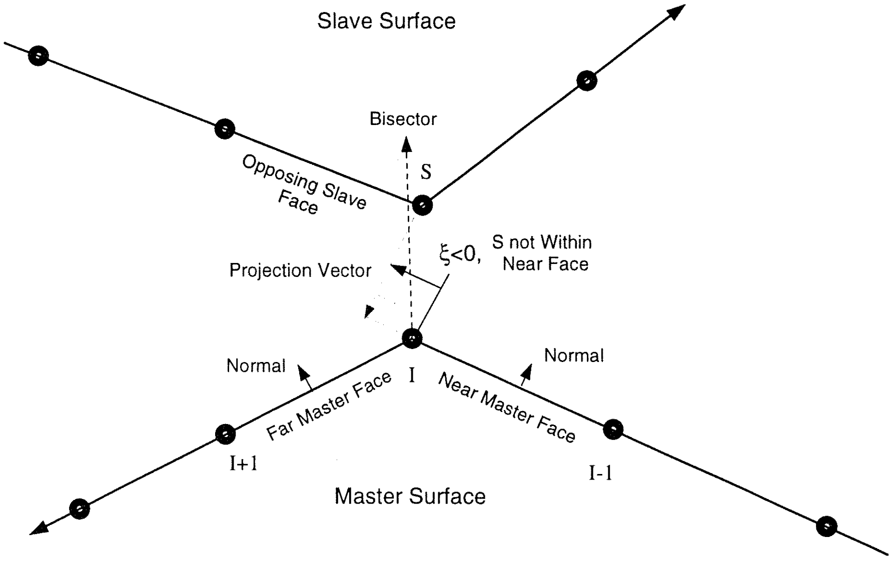

Figure 4: **Two surfaces are tested for contact**

Surface Tracking

With every time step, the contact algorithm first must predict the accelerations, velocities, and displacements for the next time step. Our version of FEAP borrows these data ($\hat{\mathbf{a}}$, $\hat{\mathbf{v}}$, $\hat{\mathbf{u}}$,) from the explicit integration routine that precedes it. The resulting displacements are then used to determine whether or not contact has taken place.

During each time step, one of the surfaces is designated the *master* surface, and the other the *slave*. The algorithm assumes slave nodes can be penetrating the faces between the master nodes. Therefore, an array is updated every time step by moving from a previous closest master node to a closer, adjacent master node, then repeats until neither adjacent node is closer. This method allows gradual relative motion between the two surfaces, while reducing the chances of sudden jumps across the material near sharp corners.

Determining Contact

Once the appropriate master node is chosen, the position of the slave node is compared to the faces on either side of the master node. For a 2-D implementation, these two faces are the lines connecting the closest master point and the master nodes preceding and following it, respectively. The slave node is projected onto each of these lines to determine whether or not it falls within the influence of either or both of the two faces.

First of all, the angle between the master faces is bisected, and the location of the slave node with respect to the bisector is used to determine which master face is *near*,

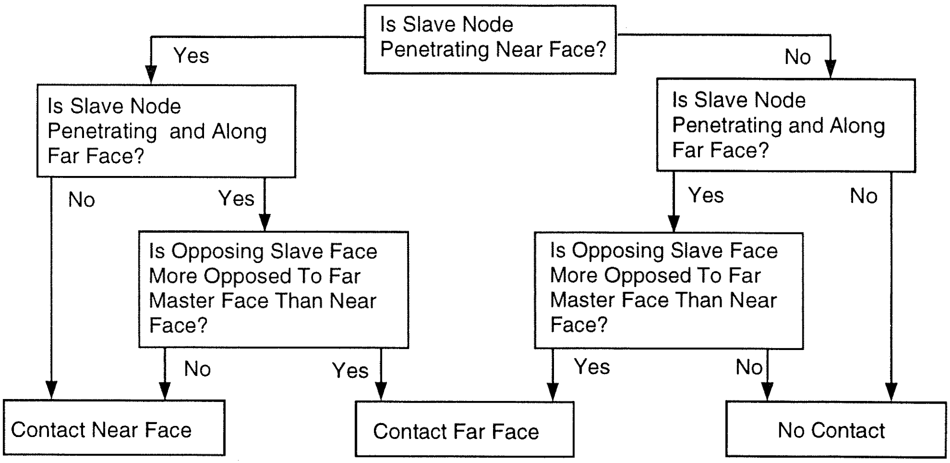
Is Slave Node Penetrating Near Face?
Yes
No
Is Slave Node Penetrating and Along Far Face?
No
Yes
Is Slave Node Penetrating and Along Far Face?
Yes
No
Is Opposing Slave Face More Opposed To Far Master Face Than Near Face?
No
Yes
Is Opposing Slave Face More Opposed To Far Master Face Than Near Face?
Yes
No
Contact Near Face
Contact Far Face
No Contact

Figure 5: **Contact determination algorithm.**

and which is *far*. The *near* face is the one which lies on the same side of the bisector as the slave node. Several tests are now made to determine if contact has occurred, as detailed in Figures 4 and 5. They hinge upon several definitions.

The slave node is considered to be *penetrating* a face if the dot product between the projection vector from the slave node to the surface and the master surface normal is positive. The slave node is considered to be *along* a face if the above projection vector intersects the line defined by the face on the same side of the angle bisector as the face itself. The *opposing slave segment* is the face on the slave surface (adjacent to the slave node) that has the most negative dot product of its normal with respect to the master face normals. Finally, the opposing slave segment is *more opposed* to the master face that supplies the most negative dot product of the normals.

Acceleration Corrections

To ensure that the two surfaces do not interpenetrate, any penetrating slave node and its associated masters must have accelerations applied to them to negate the predicted contact. First, the penetration force of the slave node is calculated by

$$\mathbf{f}_p = -\frac{\beta m_s \delta}{\Delta t^2}\mathbf{n} \tag{41}$$

in which m_s is the mass of the slave node, δ is the penetration distance, Δt is the time step, and $\mathbf{n}$ is the master surface normal vector. Parameter $0 < \beta < 1$ represents the fraction of a time step that a surface spends as a slave or master surface. For best results, similar materials should spend approximately equal amounts of time as slave and master surfaces, as master surfaces behave more stiffly than the slave surface.

Next, this force is balanced to the master nodes using the variable ξ, distance from the first node of the contact face to the projection point, nondimensionalized by the length of the face. Once this is complete, the contributions are summed and virtual work is used to generate the acceleration correction for the master nodes. The summation is over the contributions from the slave nodes contacting master node I.

$$\begin{aligned} m_{s1} &= (1-\xi)m_s && (42)\\ m_{s2} &= \xi m_s && (43)\\ \mathbf{f}_{s1} &= (1-\xi)\mathbf{f}_p && (44)\\ \mathbf{f}_{s2} &= \xi\mathbf{f}_p && (45)\\ (m+\sum_s m_{sI})\mathbf{a}_{\mathbf{n}I} &= \sum_s \mathbf{f}_{sI}. && (46) \end{aligned}$$

The correction for the slave node is then calculated using the master node responses, and is in turn used to get the final normal accelerations in the absence of friction for the slave and master nodes.

$$\begin{aligned} \mathbf{a}_{ns} &= (1-\xi)\mathbf{a}_{n1} + \xi\mathbf{a}_{n2} && (47)\\ \mathbf{a}_{slv} &= \hat{\mathbf{a}} + \mathbf{a}_{ns} - \frac{\mathbf{f}_p}{m_s} && (48)\\ \mathbf{a}_{mstr} &= \hat{\mathbf{a}} + \mathbf{a}_{nI}. && (49) \end{aligned}$$

Friction

Friction forces are handled by a velocity-dependent model, in which they resist the relative in-plane motion of the contacting surfaces. The relative velocity between the slave node and the corresponding master surface is calculated by

$$\hat{\mathbf{v}}_r = \hat{\mathbf{v}}_S - (1-\xi)\hat{\mathbf{v}}_1 - \xi\hat{\mathbf{v}}_2 \qquad (50)$$

wherein $\hat{\mathbf{v}}_S$, $\hat{\mathbf{v}}_1$, and $\hat{\mathbf{v}}_2$ are the predicticted velocities for the slave node and the two master nodes, respectively. From this, the relative tangential velocity, $\mathbf{v}_s$, its magnitude v_s, and the tangential unit vector $\mathbf{s}$ can be found by

$$\begin{aligned} \hat{\mathbf{v}}_s &= \hat{\mathbf{v}}_r - (\mathbf{n}\cdot\hat{\mathbf{v}}_r)\mathbf{n} && (51)\\ v_s &= \sqrt{\hat{\mathbf{v}}_s\cdot\hat{\mathbf{v}}_s} && (52)\\ \mathbf{s} &= \frac{\hat{\mathbf{v}}_s}{v_s}. && (53) \end{aligned}$$

Similarly to the treatment of penetration, a tangential contact force defined as a fraction of the force that must be applied to the slave node to cancel its relative motion is given by

$$f_s = -\frac{\beta m_s v_s}{\triangle t}. \tag{54}$$

A key aspect in the treatment of contact is the definition of a friction coefficient. In general the friction coefficient is a function of pressure, sliding velocity, and interface temperature. In what follows, we introduce a friction coefficient that is a function of the sliding velocity v_s, namely,

$$\mu = \mu_\infty + (\mu_o - \mu_\infty)e^{-\gamma v_s} \tag{55}$$

in which μ_0 and μ_∞ are the low and high-velocity friction coefficients, and γ is a decay constant. Additional acceleration corrections are then calculated by

$$\begin{aligned}
a_s &= -min(\mu(\mathbf{a}_{ns} - \frac{\mathbf{f}_p}{\mathbf{m}_s}) \cdot \mathbf{n}, \frac{\beta v_s}{\triangle t}) && (56)\\
\mathbf{a}_{slv} &= \mathbf{a}_{slv} + a_s\mathbf{s} && (57)\\
\mathbf{a}_{mstr_1} &= \mathbf{a}_{mstr_1} - (1-\xi)\frac{a_s m_s}{m_1}\mathbf{s} && (58)\\
\mathbf{a}_{mstr_2} &= \mathbf{a}_{mstr_2} - \xi\frac{a_s m_s}{m_2}\mathbf{s}. && (59)
\end{aligned}$$

A Dynamic Interface/Contact Model

Recently, Espinosa & Lu, 1995, proposed an extension of the above contact algorithm to model the dynamic behavior of interfaces. The model is based on the interface model proposed by Tvergaard, 1990, for quasi-static calculations and assumes that the interface carries forces that oppose separation and shear between two surfaces until debonding. The magnitude of these forces are a function of the relative separation and shear displacement between the two surfaces. The normal and tangential displacement jumps u_n and u_t, respectively, are used to determine a nondimensional parameter λ, defined as

$$\lambda = \sqrt{\left(\frac{u_n}{\delta_n}\right)^2 + \left(\frac{u_t}{\delta_t}\right)^2}, \tag{60}$$

where δ_n and δ_t are the relative displacement components at which interface separation or debonding takes place. Then, the interface traction vector may be calculated by

$$T_n = \frac{u_n}{\delta_n}F(\lambda), \quad T_t = \alpha\frac{u_t}{\delta_t}F(\lambda) \tag{61}$$

$$F(\lambda) = \frac{27}{4}\sigma_{max}(1 - 2\lambda + \lambda^2), \quad \text{for} \quad 0 \leq \lambda \leq 1, \tag{62}$$

where σ_{max} is the maximum strength of the interface. Note that total debonding occurs when $\lambda = 1$.

This interface model is included as part of an overall surface contact algorithm through the use of interface elements having nodal equivalent forces given by the above defined tractions. Two possibilities need to be considered in the calculation of nodal equivalent forces. The first is the case of separation, $u_n \geq 0$, where the two surfaces are pulling away from each other. In this case, the interface law described above is used to apply the appropriate equivalent nodal forces obtained by integration of the traction vector. If interface failure is detected, $\lambda = 1$, the surfaces can separate freely. The second case is that of penetration, $u_n \leq 0$. In this case, the contact algorithm provides the equivalent normal nodal forces, while the interface law provides the equivalent nodal tangential forces. Upon interface failure, equivalent nodal tangential forces are computed from the friction law.

THERMAL EFFECTS

As stated in the introduction, large plastic strains and strain rates result in temperature increases in the body. Hence a temperature-dependent material model and heat conduction need to be added to account for the fact that the behavior of most materials can change dramatically as the temperature rises. The rise in temperature is usually caused by the generation of heat through plastic deformation but can also be the result of friction between sliding interfaces. In this section a discretized heat equation and its temporal integration is discussed. Moreover, an algorithm accounting for thermal conductance between bodies in contact consistent with the treatment of mechanical contact, is presented.

Governing Equations For Thermal Effects

In a temperature-dependent model, an extra level of complexity is added to the governing equations. Heat production and transfer is controlled by

$$k\underbrace{(T_{,xx}+T_{,yy})}_{\nabla^2 T}+Q = c\rho\dot{T} \tag{63}$$

in volume B_t, where heat is generated and transferred. In the above equation, Q is the heat source, ρ is the mass density, c is the specific heat, and k is the thermal conductivity.

Boundary conditions consist of surface S_q, upon which a prescribed heat flux q_B is applied, surface S_c, upon which proscribed heat convection $h(T_f - T)$ is applied, and surface S_T, upon which a given temperature is prescribed. These conditions are given for each surface as

$$\begin{aligned} T &= T_B \quad \text{on} \quad S_T && (64)\\ q &= q_B = k(T_{,x}n_x + T_{,y}n_y) \quad \text{on} \quad S_q && (65)\\ q &= h(T_f - T) = k(T_{,x}n_x + T_{,y}n_y) \quad \text{on} \quad S_c && (66)\end{aligned}$$

where T_f is the temperature at infinity, T is the temperature on S_c, and $\mathbf{n} = (n_x, n_y)$

is the surface normal vector. A weak form of 64, 65, 66 is given by

$$\int_{B_t}(k\nabla^2T+Q-c\rho\dot{T})\eta dB-\int_{S_q}(k\nabla T\cdot\mathbf{n}-q_B)\eta dS-\int_{S_c}(k\nabla T\cdot\mathbf{n}-h(T_f-T))\eta dS=0 \tag{67}$$

where $S = S_T \cup S_c \cup S_q$, and η is a test function satisfying the boundary condition $\eta = 0$ on S_T. It can be shown that an equivalent expression of the weak form is

$$\int_{B_t} k(\nabla T\cdot\nabla\eta)dB+\int_{S_c} hT\eta dS+\int_{B_t} c\rho\dot{T}\eta dB=\int_{S_q} q_B\eta dS+\int_{B_t} Q\eta dB+\int_{S_c} hT_f\eta dS \tag{68}$$

This equation expresses an energy balance wherein the change in heat energy in volume B_t is equal to the net heat added.

For a finite element approach, a Galerkin method provides the discretization

$$\begin{aligned}
\mathbf{T}(\mathrm{x}) &= \mathrm{N}(\mathrm{x})T_e \qquad (69)\\
\eta(\mathrm{x}) &= \mathrm{N}(\mathrm{x})\eta_e \qquad (70)\\
\mathbf{T}_\partial &= \mathbf{B}\mathbf{T}_e \qquad (71)
\end{aligned}$$

in which $\eta(\mathrm{x})$ is a weight function approximated using the same shape functions N as used for T. Note that these shape functions are defined in the deformed configuration. The quantity T_e is the array of nodal temperatures for an element, while T_∂ is the matrix containing the partial derivatives of T_e across the element.

Given the above, the governing equations simplify into the finite element equation

$$(\mathbf{K}+\mathbf{H})\mathbf{T}+\mathbf{C}\dot{\mathbf{T}}=\mathbf{R}_\mathrm{q}+\mathbf{R}_\mathrm{Q}+\mathbf{R}_\mathrm{h} \tag{72}$$

wherein the global characteristic conduction, convection, and rate dependent matrices $\mathbf{K}$, $\mathbf{H}$, $\mathbf{C}$ determine the response of the system. They may be compiled from the element matrices by

$$\begin{aligned}
\mathbf{K} &= \sum_e \mathrm{k_e} = \sum_e \int_{A_e} \mathrm{B}^T\mathbf{kB}dA \qquad (73)\\
\mathbf{H} &= \sum_e \mathrm{h_e} = \sum_e \int_{S_c} \mathrm{N}^T h\mathrm{N}dS \qquad (74)\\
\mathbf{C} &= \sum_e \mathrm{c_e} = \sum_e \int_{A_e} \mathrm{N}^T \rho c\mathrm{N}dA \qquad (75)\\
\mathbf{R}_\mathrm{q} &= \sum_e \int_{S_B} \mathrm{N}^T q_B dS \qquad (76)\\
\mathbf{R}_\mathrm{h} &= \sum_e \int_{S_c} \mathrm{N}^T T_f h dS \qquad (77)\\
\mathbf{R}_\mathrm{Q} &= \sum_e \int_{A_e} \mathrm{N}^T Q dA \qquad (78)
\end{aligned}$$

$$= \sum_e (\sum_{p=1}^{NQP} W_p \mathbf{N}_p^T Q_p) \tag{79}$$

where $\mathbf{R_q}$, $\mathbf{R_h}$, and $\mathbf{R_Q}$ are the thermal load vectors.

If thermal conductivity is assumed to be a crystal property, then the large displacements and rotations introduced by the elastic part of the deformation gradient needs to be accounted for. Following Marusich and Ortiz, 1995, we define $\mathbf{k} = k\mathbf{F^e}\mathbf{F^{eT}}$ in 96. Furthermore, for the plastic process defined earlier, a heat supply Q due to the rate of plastic work, $\dot{W}^p$, can be estimated by the Taylor-Quinny formula, viz.,

$$Q = \alpha \dot{W}^p \tag{80}$$

in which α is a coefficient of the order of 0.8-0.9.

The whole system of equations may be simplified by joining $(\mathbf{K} + \mathbf{H}) = \hat{\mathbf{K}}$, so that

$$\hat{\mathbf{K}}\mathbf{T} + \mathbf{C}\dot{\mathbf{T}} = \mathbf{R}. \tag{81}$$

Now, using a direct integration method, the temperature field is computed at each time step as

$$(\frac{1}{\triangle t}\mathbf{C} + \beta\hat{\mathbf{K}})\mathbf{T}_{n+1} = (\frac{1}{\triangle t}\mathbf{C} - (1-\beta)\hat{\mathbf{K}})\mathbf{T}_n + (1-\beta)\mathbf{R}_n + \beta\mathbf{R}_{n+1} \tag{82}$$

so that, if $\beta = 0$, the process becomes an explicit integration algorithm,

$$\mathbf{C}\mathbf{T}_{n+1} = (\mathbf{C} - \triangle t\hat{\mathbf{K}})\mathbf{T}_n + \mathbf{R}_n. \tag{83}$$

This assumes, of course, initial conditions $\mathbf{T} = \mathbf{T}_o$ at $t = 0$, as well as a time step below $\triangle t_{cr} = 2/\omega_{max}$. In addition, if $\mathbf{C}$ is lumped the algorithm does not require factorization of $\mathbf{C}$.

Thermal Contact Conductance

The objective is to formulate an algorithm that can account for heat fluxes between solid bodies. We derive such a scheme consistently with the treatment of mechanical contact, i.e., we define master and slave surfaces each having a different temperature. If the mechanical contact determines penetration, we define a heat flux between the surfaces due to a temperature jump and heat generated at the frictional contact resulting from a jump in velocities across the sliding surface. This heat flux is given by

$$q_p = \beta[h_s(T_s - T_m) - \mathbf{t} \cdot (\mathbf{v_s} - \mathbf{v_m})] \tag{84}$$

in which $\mathbf{t}$ is the contact traction vector, $\mathbf{v_s}$ and $\mathbf{v_s}$ are the slave and master nodal velocity vectors, T_s is the temperature of the slave node and T_m is the temperature of the master surface at the projection point, viz.,

$$T_m = \xi T_2 + (1-\xi)T_1, \tag{85}$$

where T_1 and T_2 are the temperatures of the precedent and antecedent nodes, respectively, on the master segment. The contact heat transfer coefficient h_s is a function of surface roughness L_g, contact area fraction f_c, material conductivity of the slave and master surfaces k_s and k_m, the conductivity of the material filling the interface k_f, and void area fraction f_v, Özisik, 1977,

$$h_s = \frac{1}{L_g}(f_c \frac{2k_s k_m}{k_s + k_m} + f_v k_f). \tag{86}$$

In general $k_f \ll k_s, k_m$ so the last term in the above equation can be neglected. The contact area fraction generally depends on the interface pressure. We propose to use an exponential law for f_c given by

$$f_c = 1 - f_1 e^{-\alpha p}, \tag{87}$$

where f_1 and α are parameters defining the initial area fraction and the rate at which f_c approaches unity with pressure.

Next, the heat flux is distributed in both surfaces consistently with the discretized formulation of the heat equation. Equivalent nodal forcing terms are computed by

$$\mathbf{r}_q^s = \int_{A_s} q_p \mathbf{N} dA \tag{88}$$

where A_s is the slave node influence surface area. Temperature changes in slave and master nodes are given by

$$(C_I + \sum_s C_{sI})\Delta T_I = \sum \mathbf{r}_{qI}^s \Delta t \tag{89}$$

$$T_I = \hat{T}_I + \Delta T_I \tag{90}$$

$$\Delta T_s = (1-\xi)\Delta T_1 + \xi \Delta T_2 \tag{91}$$

$$T_s = \hat{T}_s + (\Delta T_s - \Delta \hat{T}). \tag{92}$$

It should be noted that in the present algorithm no assumption implying continuity of the temperature field across the contact surface is made. Moreover, thermal contact conductance and frictional heat sources are naturally incorporated in the mechanical contact algorithm. The scheme is easily implemented as part of the contact algorithm discussed previously.

ADAPTIVE MESH REFINEMENT

Element distortion in dynamic finite plasticity calculations can reduce the stable time step for an explicit method to a point where the computation no longer advances and the field variables are not accurately interpolated. A solution to this problem is to rediscretize the domain with a new, undistorted mesh, and continue with the calculation. This requires a procedure to transfer the problem state variables to the new mesh, while satisfying the governing equations, such that the calculation may continue without excessive error.

Update of Configuration

Central to the idea of replacing an old, highly distorted mesh, with a new, undeformed mesh is the update of the reference configuration. The model presented earlier, governing the material behavior, uses a Lagrangian formulation. If a mesh is rebuilt, but the reference configuration is *not* updated, the distortion merely changes it's location from the current configuration to that of the reference. To truly remove the distortion, the reference configuration must be moved forward to the current configuration.

The reference configuration is defined as the configuration at which the displacements are zero. So part of updating the reference configuration is setting

$$\begin{aligned} \hat{\mathbf{x}} &= \mathbf{X} + \mathbf{u} && (93) \\ \hat{\mathbf{u}} &= \mathbf{0} && (94) \\ \mathbf{x} &= \hat{\mathbf{x}} + \hat{\mathbf{u}} && (95) \end{aligned}$$

wherein $\mathbf{X}$, $\mathbf{u}$, $\hat{\mathbf{x}}$, and $\hat{\mathbf{u}}$ are the reference particle positions and displacements in the old and new reference configurations, respectively. In addition, after the update the new current particle position $\mathbf{x}$ is defined by the new reference $\hat{\mathbf{x}}$ and displacement $\hat{\mathbf{u}}$, as depicted in Figure 6. More is required, however. Remember that the deformation gradient relates the current configuration to the reference configuration by

$$\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} = \frac{\partial(\mathbf{X} + \mathbf{u})}{\partial \mathbf{X}} = \mathbf{I} + \frac{\partial \mathbf{u}}{\partial \mathbf{X}} \qquad (96)$$

where $\mathbf{x}$ is the current position field. When the reference is updated, however, the relation becomes a bit more complicated. After an update, the deformation gradient expands to

$$\begin{aligned} \mathbf{F} &= \frac{\partial \mathbf{x}}{\partial \mathbf{X}} = \frac{\partial(\mathbf{X} + \mathbf{u} + \hat{\mathbf{u}})}{\partial \mathbf{X}} && (97) \\ &= \mathbf{I} + \frac{\partial \mathbf{u}}{\partial \mathbf{X}} + \frac{\partial \hat{\mathbf{u}}}{\partial \hat{\mathbf{x}}} \frac{\partial \hat{\mathbf{x}}}{\partial \mathbf{X}} && (98) \\ &= (\mathbf{I} + \frac{\partial \mathbf{u}}{\partial \mathbf{X}}) + \frac{\partial \hat{\mathbf{u}}}{\partial \hat{\mathbf{x}}}(\mathbf{I} + \frac{\partial \mathbf{u}}{\partial \mathbf{X}}) && (99) \end{aligned}$$

such that $\partial \mathbf{u}/\partial \mathbf{X}$ may be stored and reset at each update as

$$\frac{\partial \mathbf{u}}{\partial \mathbf{X}} = \mathbf{F} - \mathbf{I}. \qquad (100)$$

We can then compute the deformation gradient $\mathbf{F}$ as

$$\begin{aligned} \mathbf{F}^{old} &= \mathbf{I} + \frac{\partial \mathbf{u}}{\partial \mathbf{X}} && (101) \\ \mathbf{F} &= \mathbf{F}^{old} + \frac{\partial \hat{\mathbf{u}}}{\partial \hat{\mathbf{x}}} \frac{\partial \hat{\mathbf{x}}}{\partial \mathbf{X}}. && (102) \end{aligned}$$

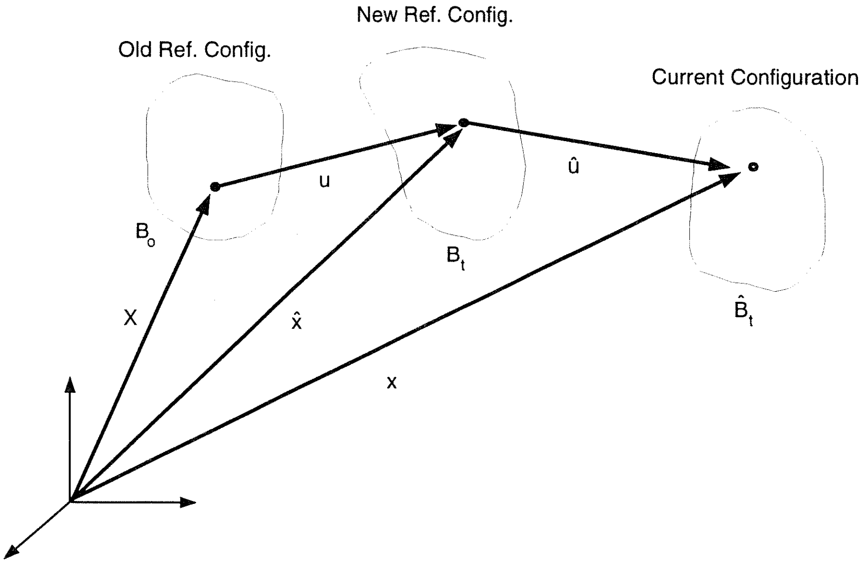

Figure 6: **Update of current configuration.**

In discretized form, equation 99 becomes

$$
\begin{aligned}
F_{iJ} &= \sum_{a}^{NEN} N_{a,J}\hat{x}_{ia} & (103)\\
&= I_{iJ} + \sum_{a}^{NEN} N_{a,J}u_{ia} + \sum_{a}^{NEN} \hat{N}_{a,k}\hat{u}_{ia}(I_{kJ} + \sum_{b}^{NEN} N_{b,J}u_{kb}) & (104)
\end{aligned}
$$

where NEN is the number of nodes per element. Shape functions derivatives, $N_{a,J}$ and $\hat{N}_{a,k}$, are defined with respect to the original and new reference configurations, respectively.

Interpolation and Consistent Computation of Field Variables

The determination of field variables in the new mesh is accomplished by interpolating the quantities in the deformed configuration based upon the shape functions of an auxiliary element connectivity of simple 3-node triangles connecting all of the existing nodes in the old mesh. For each node in the new mesh, the area coordinates of that node are computed for each auxiliary element connecting nodes in the old mesh. These area coordinates are then used as shape functions for the interpolation of the nodal variables. This sort of interpolation is conducted for the nodal kinematic quantities, including displacement, velocity, and acceleration. In addition, the same sort of interpolation is performed for the components of the stress tensor $\overline{\mathbf{S}}$ and the internal variables $\overline{\mathbf{Q}}$

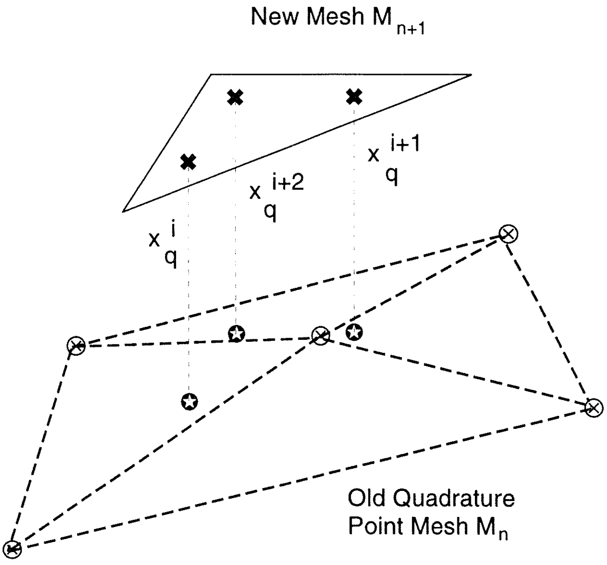

Figure 7: **Interpolation from quadrature points in the old mesh to quadrature points in the new mesh.**

Set of interpolated variables:	$\mathbf{I}(\ \mathbf{u},\ \mathbf{v},\ \mathbf{a},\ T,\ \overline{\mathbf{S}},\ \epsilon^p,\ \dot{\epsilon}^p,\ \theta,\ \mathbf{F}^{old})$
Set of computed variables:	$\mathbf{C}(\ \overline{\mathbf{C}}^e,\ \mathbf{F}^e,\ \mathbf{F}^p)$

Table 3: **Interpolated and computed state variables.**

using this time an auxiliary mesh obtained by 3-node triangular elements connecting quadrature points in the old mesh, as shown in Figure 7.

It should be noted that not all of the variables may be interpolated. If everything was interpolated, the stress state after the mesh update would be inconsistent with the state of displacement. The problem then requires a suitable choice of field variables to be interpolated and other field variables to be computed to keep the solution consistent and stable. The field variables are divided up as shown in Table 3. This is possibly the most important choice of the remeshing scheme. Remember that the deformation gradient has the multiplicative decomposition $\mathbf{F} = \mathbf{F}^e\mathbf{F}^p$. After interpolation of the set $\mathbf{I}$ the $\mathbf{F}^p$ is computed from the stresses and displacements so that any error introduced through the remeshing ends up in a compatible $\mathbf{F}^p$, instead of allowing inconsistencies to build up.

The first step in the calculation of $\mathbf{F}^p$ requires the determination of the elastic part of the deformation gradient in the old mesh. Therefore, before the interpolation is begun, the deformation gradient $\mathbf{F}$ is computed from the displacement field and the elastic part of $\mathbf{F}$, $\mathbf{F}^e$, is computed from the multiplicative decomposition of $\mathbf{F}$ at the quadrature points in the old mesh, namely

$$\mathbf{F}^e = \mathbf{F}\mathbf{F}^{-p}. \tag{105}$$

Then, through a polar decomposition of $\mathbf{F}^e$, a rotation matrix $\mathbf{R}^e$ may be identified, and rendered down to a rotation angle θ as shown below.

$$\begin{aligned} \mathbf{R}^e &= \mathbf{F}^e\mathbf{U}^{e-1} \\ \mathbf{R}^e &= \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned} \tag{106}$$

This theta is interpolated with the internal variables, then used to rebuild $\mathbf{R}^e$ for the following calculation of $\mathbf{F}^p$ in the new mesh. Once the stresses and rotation angle have been interpolated, the shape functions in the new mesh calculated, and the deformation gradient $\mathbf{F}$ computed from the interpolated displacement field $\mathbf{u}$ and $\mathbf{F}^{old}$, the plastic part of the deformation gradient $\mathbf{F}^p$ is recalculated for each quadrature point.

Before calculating $\mathbf{F}^e$, the interpolated stresses must be corrected to maintain a consistent rate of plastic strain. The current material strength $\mathbf{s}$ is updated using the interpolated effective plastic strain ϵ^p according to equation 22. In addition, the deviatoric portion of the second Piola-Kirchhoff stress $\overline{\mathbf{S}}_{dev}$ of the interpolated stress tensor is corrected using an interpolated effective plastic strain rate $\dot{\epsilon}^p$, viz.,

$$\overline{\mathbf{S}}^{corr}_{dev} = \overline{\mathbf{S}}^{int}_{dev}\frac{\mathbf{s}}{\overline{\sigma}}\left(\frac{\dot{\epsilon}^p + \dot{\epsilon}^p_o}{\dot{\epsilon}^p_o}\right)^{\frac{1}{\alpha}}, \tag{107}$$

after which the stress tensor is rebuilt with the new $\mathbf{S}_{dev}$. This is done to preserve the plastic strain rate through the mesh, as the plastic strain rate is sensitive to sudden changes in the effective stress. The elastic right Cauchy-Green deformation tensor is reconstructed from the material stiffness tensor L and the second Piola-Kirchhoff stress $\overline{\mathbf{S}}$, to produce the elastic part of the deformation gradient $\mathbf{F}^e$, namely,

$$\mathbf{C}^e = \exp(2L^{-1}\overline{\mathbf{S}}) \tag{108}$$

$$\mathbf{F}^e = \mathbf{R}^e\sqrt{\mathbf{C}^e}. \tag{109}$$

Before computing $\mathbf{F}^p$, $\mathbf{F}^e$ is scaled so that $\det\mathbf{F}^e = \det\mathbf{F}$. This is necessary because the plastic deformation is assumed to be isochoric ($\det\mathbf{F}^p = 1$), and this must be preserved through the calculation. Finally, the plastic deformation gradient is computed by

$$\mathbf{F}^p = \mathbf{F}^{e-1}\mathbf{F} \tag{110}$$

1. Define boundaries for new mesh based on old mesh.
2. Create new set of corner nodes from old mesh. Refine as necessary.
3. Create new elements by Delaunay triangulation.
4. Remove elements created outside mesh boundaries.
5. Optimize element shapes.
6. Define auxiliary interpolation meshes connecting nodes, quadrature points.
7. Perform polar decomposition on $\mathbf{F}^e$, and save orientation in θ.
8. Interpolate variables in set **I**.(See Table 3)
9. Update reference configuration.
10. Compute new shape functions.
11. Compute variables in set **C**. (See Table 3)
12. Replace old state variables with new, and continue with the calculation.

Table 4: **Field variable transfer algorithm.**

If everything is done correctly, the $\mathbf{F}^p$ after remeshing should resemble the $\mathbf{F}^p$ in the old mesh. A summary of the mesh transfer algorithm is given in Table 4.

Remeshing and Mesh Refinement Method

Remeshing Condition

Transferring the state of a calculation from one discretization to another is computationally expensive. The key, therefore, to deciding when to undergo this process is finding when the calculation is becoming inefficient enough with the old mesh to require a new one. One way to handle this is to track the stable time step for the explicit integration. As elements become more deformed, the stable time step will dwindle. A simple criterion for remeshing, therefore, is to set a threshold time step that will force the program to remesh. The condition currently in use compares the current time step to a reference value set at the end of the previous remeshing. In this way, if the time step dwindles to below a fraction of its reference value, a remeshing will be triggered. This fraction is provided by the user.

Another possible criterion for remeshing could be based on the norm of the displacement gradient $||\nabla \mathbf{u}||$ over the whole mesh. The norm may be calculated at each element, then compared to a threshold value set by the user. This condition should work well coupled with the update of the configuration, as it provides a measure of the severity of the deformation.

Refinement condition

When remeshing, it can be useful to introduce more elements in those regions of the mesh that are rapidly deforming. One way of deciding which regions need to be refined is to examine the rates of plastic work being produced in the elements, Marusich and Ortiz, 1995.

The rate of plastic work $\dot{W}^p$ is computed at each quadrature point as $\dot{W}^p = \overline{\sigma}\dot{\epsilon}^p$. These work rates are averaged for the element. The element rates are in turn averaged over the whole mesh. Each element rate is compared to the average to decide whether the element should be refined, coarsened, or left unchanged. This will depend on user-supplied thresholds, as in

$$\begin{aligned} I_e &= \sum_{i=1}^{NQUAD} \dot{W}_i^p / NQUAD & (111) \\ I &= \sum_{e=1}^{NUMEL} I_e / NUMEL & (112) \\ \beta_e &= I_e / I & (113) \end{aligned}$$

$$\begin{array}{ll} \beta_e > \beta_{max} & \text{needs to refine} \\ \beta_e < \beta_{min} & \text{needs to coarsen,} \end{array} \qquad (114)$$

where β_{max} is a threshold number greater than one, and β_{min} is between zero and one. In addition, maximum and minimum element sizes are defined to limit refinement and coarsening that could lead to excessive computational times or poor accuracy.

Element Refinement

When a remeshing is triggered, the new mesh is created by a delaunay triangulation algorithm, redefining the element connectivity using the current locations of all of the corner nodes from the old mesh. When an element is flagged for refinement, its midnodes are added to the corner node list, resulting in several elements being created where just one existed before. This method for refinement is used primarily because it is a very simple and easy way to introduce new elements. Even those elements which are not refined or optimized are improved in the remeshing operation. As the delaunay triangulation routine creates new 6-node elements from existing corner nodes, it attempts to connect them so that the elements are as close to equilateral as convenient. In addition, new midnodes are created at the midpoints of the element sides, so that new elements are perfectly triangular.

Other Considerations

Of course, mesh refinement requires several other considerations that greatly depend on the specifics of the problem being studied. For example, the boundary conditions would need to be set for the new mesh and the nodes on the surfaces of bodies redefined for contact. In addition, in a multi-body problem, it is possible that for some of the bodies, remeshing is neither needed nor desired. These are problems that may be considered on a case-by-case basis.

Mesh Optimization

Not all meshes are suitable for finite element calculations. The error due to the approximations depends on the size and the shape of the elements. The element sizes are chosen as a compromise between accuracy and computing cost, and may vary through the domain according to error estimates or user's experience. Good meshes may be set apart from bad ones by the shapes of the elements contained within. Distorted elements produce errors which degrade the quality of the calculation, and elements with a good initial shape may distort as they follow the movement of the domain's boundary. This has motivated intensive research on mesh-improving techniques, and much experience in smoothing meshes has come from the field of deformable domains. In Zavattieri et al., 1995, it was proposed to use, as quality of the mesh, the quality of its *worst* element, defined by a measure Q,

$$Q_{global} = \min_k Q_k \tag{115}$$

This choice incorporates a well-known fact that one unacceptable element renders the mesh useless. The problem then becomes one of maximizing an objective function of the min type. As min functions are non-differentiable, a simple node-by-node algorithm (described later) is used. This implementation uses a node-movement algorithm that, in most cases, produces a sequence of meshes of non-decreasing quality.

A Global Mesh Optimization Algorithm

The simultaneous optimization of the locations of all of the nodes in a mesh consisting of thousands of elements is obviously untractable. In Zavattieri, et al., 1995, it is shown that to improve the worst element of the mesh, it is only necessary to move those nodes located in close vicinity to it (up to its neighbors of order 1 or 2). Also, in Zavattieri, et al., 1995, they have incorporated a computationally inexpensive measure of the quality of a triangle, namely,

$$Q_e = \mathrm{C}\frac{A_e}{\mathcal{P}_e^2} \tag{116}$$

where A_e is the area of element e, $\mathcal{P}_e$ its perimeter (the sum of the lengths of its three edges), and C is a constant which renders the quality of an equilateral triangle equal to one ($C = 20.784619$). The advantages of this definition of quality is discussed in Zavattieri, 1995.

The objective function Q_{global} is maximized over the space of nodal locations. We only modify the positions of those nodes that belong to the worst element of the mesh, and their nth-order neighbors (with n=1 or 2). We will refer to as *moving nodes* those nodes that are moving at a given stage of the algorithm. The *moving nodes* are selected among the *movable nodes.* In the present version of our method, boundary nodes are *non-movables.* If all the nodes belonging to an element are *non-movables*, the element is not considered. The algorithm which identifies nodes as *moving*, *movable* or *non-movables* is displayed in Table 5. The current implementation uses NL=2, and

1. Initialization and Global Parameters: Specify a neighborhood level NL and a maximum number of iterations M. Set boundary nodes as non-movable. Initialize an integer auxiliary constant PREV with 0.
2. Identify the worst element among those elements of the mesh containing at least one *movable node*, KWORST. If the number of *movable nodes* is zero, stop.
3. If KWORST=PREV, set the three corner nodes of KWORST as *non-movable* and go back to 2.
4. Identify the nodes in a neighborhood of order NL of KWORST. The neighbors of order 0 are the four nodes belonging to KWORST. Neighbors of order 1 are those nodes that are not neighbors of order 0, bur are connected by an edge with at least one neighbor of order 0; and so on.
5. Set as *moving nodes* those identified in 3 that have not previously been set as *non-movable*. Sweep these *moving nodes* modifying the nodal positions one at a time, according to the *local cluster optimization* rule (see below), so as to improve the quality of the cluster.
6. Assign to PREV the value KWORST.
7. Go back to 2, or Stop if the number of iterations is equal to M.

Table 5: **Identification of moving nodes.**

M=$NUMEL/3$ where $NUMEL$ is the number of element of the mesh.

Local Cluster Optimization

The nodal position of each node I is modified to maximize the quality Q_e of the elements that share this node. Let us define the *cluster* C_I associated to a node I of a triangulation as the set of elements that share this node. We associate to the cluster C_I a scalar quantity, the quality of the cluster, defined by

$$q_C = \min_{e \in C_I} Q_e. \tag{117}$$

Quality q_C is continuous function of the position x_I of the central node I of the cluster. We next introduce a set of sampling positions. Let x_I^0 be the original position of node I. Then, for every vertex V of the cluster define two sampling points located at

$$x_{sp} = \pm \alpha x_v + (1 \mp \alpha) x_I^0, \tag{118}$$

where α is a small parameter, typically 0.05. Notice that x_I^0 is itself a sampling position. In two dimensions, the set of sampling points defined in this way would look much as in Figure 8.

The local cluster optimization rule that has been implemented is the following:

Evaluate q_I for all sampling positions, and choose as updated position the sampling position that maximizes this quality.

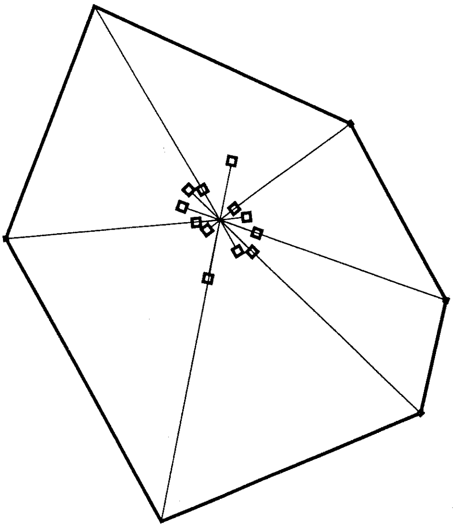

Figure 8: **Sampling positions for local cluster optimization in 2D.**

Most clusters with bad initial quality attain very high quality in just a few iterations, and then spend lots of local Iterations improving the quality from, say, 0.4 to 0.45. A good implementation should consider this possibility and stop the local Iterations whenever a cluster has reached a quality that is greater that some given threshold value QTol, so as to avoid wasting CPU time improving already-good clusters, and turn to worse ones. In the calculations later reported a value of $QTol = 0.9$ was used.

AN APPLICATION TO ROD ON ROD IMPACT.

One important objective in developing a finite element model accounting for large deformation plasticity and brittle failure is the ability of simulating specially designed experiments towards the identification of material inelasticity and failure mechanisms. One such experiment involves the collision of a 6061-T6 aluminum anvil with an unconfined AD-99.5 alumina rod in a modified Taylor test, Wise and Grady, 1994. In this test, interferometric measurement of radial and axial velocities provides an estimate of the compressive dynamic yield strength of the ceramic rod under uniaxial stress. In Wise and Grady experiments, rod free end velocity was measured with a velocity interferometer (VISAR). Unlike the Taylor test, which was extensively used in the high strain rate characterization of metals, analyses of the experimentally recorded velocity histories do not rely on any simplifying assumptions regarding rod deformation, and do not require an intact rod for recovery. This makes the modified Taylor test very useful for the study of ceramics failure through numerical simulation of the failure event.

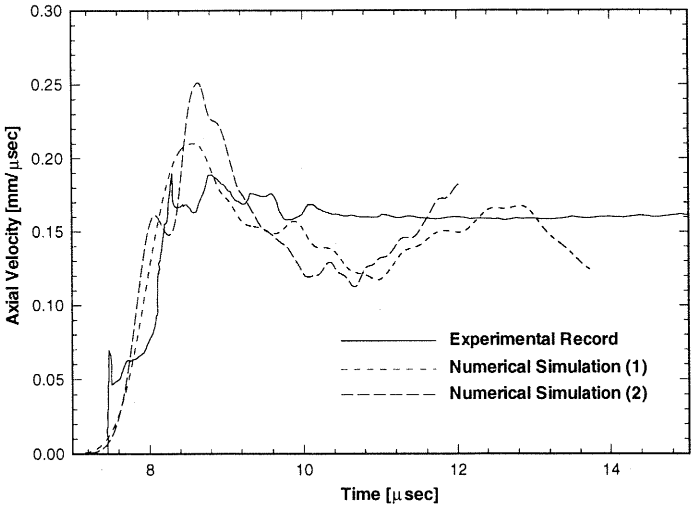

Figure 9: **Axial velocity on alumina bar free surface.**

Problem Definition

For this problem, the aluminum impactor is formed into a 134.7mm long T-section rod with diameters of 20mm and 89mm for the rod and T-flange, respectively. The flange is 12.7mm thick, while the trailing rod is 122mm long. The rod section of the anvil was initially incorporated by Wise and Grady to maximize the duration of high stress levels at the impact end of the target rod. The ceramic rod being tested is an AD-99.5 alumina rod 80mm long, with a 10.06mm diameter.

In our simulations, the anvil is modeled with 6-node axisymmetric triangular elements using the large deformation material model presented earlier is subjected to mesh refinement. The ceramic rod also uses 6-node axisymmetric triangular elements, but this time based on the small-strain microcracking multiple-plane model previously discussed. No remeshing or optimization is done for the alumina rod. The properties of the two materials being used are displayed in Table 5 below. In the calculation featured here, the anvil strikes the specimen at 1035 m/s, in an attempt to simulate the conditions for shot 5 in Wise and Grady's experiments.

Velocity Histories

During the impact calculation, the axial velocity at the free end of the ceramic rod was recorded for comparison with the experimental results of Wise and Grady. In the experimental records, as seen in Figure 9, the velocity curve is shaped by the arrival of an initial compression, followed by a second compression wave that brought the surface

6061-T6 Aluminum.	
Elastic Properties	
$E = 71.6$ GPa	Young's modulus
$\mu = 0.3414$	Poisson's ratio
$\rho_o = 2700$ kg/m^3	Density
Inelastic Properties	
$\sigma_o = 270$ MPa	Initial yield stress
$\epsilon^p_o = 0.003913$	Reference plastic strain
$\dot{\epsilon}^p_o = 1000$ s^{-1}	Reference plastic strain rate
$\alpha = 3$	Rate sensitivity exponent
$\beta = 15$	Hardening exponent
AD-99.5 Alumina Ceramic.	
Elastic Properties	
$E = 374$ GPa	Young's modulus
$\mu = .22$	Poisson's ratio
$\rho_o = 3890$ kg/m^3	Density
Inelastic Properties	
$\nu = 0.1$	Internal friction coefficient
$c_R = 5000$ m/s	Wave speed
$K_{IC} = 1.7 \times 10^6$	Critical stress intensity factor
$a_o = 10\mu m$	Initial crack radius
$c^1_n = 1.0 \times 10^{12}$ m^{-3}	Crack density for plane 1
$c^2_n = 1.0 \times 10^{12}$	Crack density for plane 2
$c^3_n = 0.0$	Crack density for plane 3
$c^4_n = 0.0$	Crack density for plane 4
$c^5_n = 0.0$	Crack density for plane 5
$c^6_n = 5.0 \times 10^{10}$	Crack density for plane 6
$c^7_n = 0.0$	Crack density for plane 7
$c^8_n = 5.0 \times 10^{10}$	Crack density for plane 8
$c^9_n = 0.0$	Crack density for plane 9
Contact properties	
$\mu_o = .5$	Static friction coeff.
$\mu_\infty = .5$	Dynamic friction coeff.
$\gamma = 0.$	Friction lawe exponent
$\beta = .5$	Contact weighting parameter

Table 6: **Material properties for aluminum/alumina collision.**

velocity to its highest value on the order of .20 mm/μs, according to Wise & Grady, 1994. The first wave travels at the ultrasonic longitudinal wave speed $C_L = 10.6$mm/μs, while the second propagated at the calculated bar wave speed $C_B = 9.81$mm/μs. After the velocity peaked, it decayed by about 20% to a relatively constant velocity. Two numerical simulations, with coarse (1) and fine (2) meshes, are plotted in Figure 9. The calculated free surface velocity, in numerical simulation (1), also rises quickly to a peak on the order of .20 mm/μs, although not as quickly as Wise and Grady's curve. This retarded arrival is likely due to the dispersion caused by the relatively coarse mesh currently modeling the ceramic. In the process, this dispersion eliminates the initial compression wave. When a finner mesh is used, the first compressive pulse appears in the velocity trace, although its magnitude is about twice the magnitude of the experimentally recorded first compressive pulse. After the peak, the velocity decays to a mean value that is about 75% of the peak value. Notable in the decay is an overshoot, which then corrects. Espinosa and Brar, 1995, showed that the almost constant axial velocity, at the ceramic bar end, is the result of ceramic spallation upon wave reflection at the ceramic rod free surface.

The lack of a well defined elastic precursor in the numerically obtained axial velocity history, needs further investigation. It may be due, at least in part, to insufficient inelasticity in the ceramic model. In the present model, only microcracking is considered. It was shown by Espinosa et al., 1992, that the glassy phase present in the ceramic microstructure, can lead to a viscoplastic like behavior that can result in further wave front attenuation (precursor decay).

Contours of Field Variables and Mesh Evolution

To illustrate the features included in the computational algorithm, mesh refinement and optimization, and mechanical contact, a series of contour/mesh plots showing the state of the finite element calculation every 2 microseconds is presented. Figures 10-15 are a series of contour plots showing the axial velocities throughout the two rods every 2 microseconds. As time progresses, we can see the stress wave move up the alumina rod (the upper rod) at a velocity of about 10,000 m/s. Notice that the alumina rod behaves in a more stiff manner than the plastically-deforming aluminum, up until around 12 microseconds after impact. Then the elements at the end of the alumina rod reach a point in their state of cracking where they are essentially pulverized, and lose nearly all stiffness.

As can be expected, the aluminum anvil is cratered by the impact with the ceramic specimen, and provides a good test for the contact and remeshing capabilities of the algorithms presented in this paper. The contact algorithm implemented in this calculation allows the aluminum to flow around the penetrating ceramic rod, gradually tracking the surfaces and redefining closest master nodes to reflect the physical problem. As deformation increases, the mesh refines itself six times automatically, using the

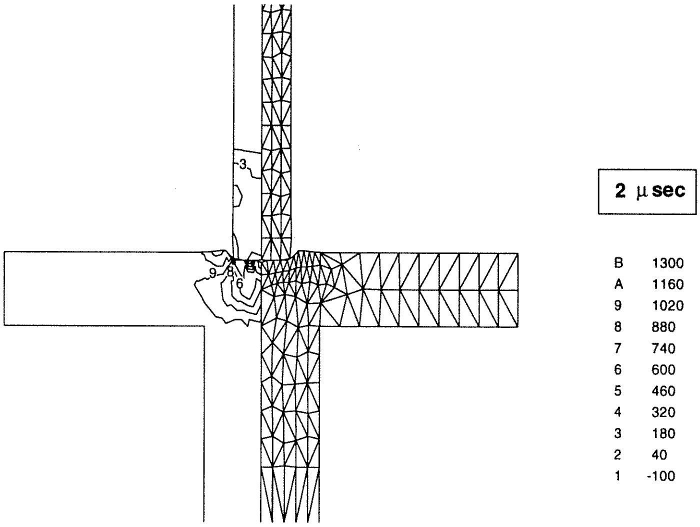

Figure 10: **Axial velocity contours at t=2μs.**

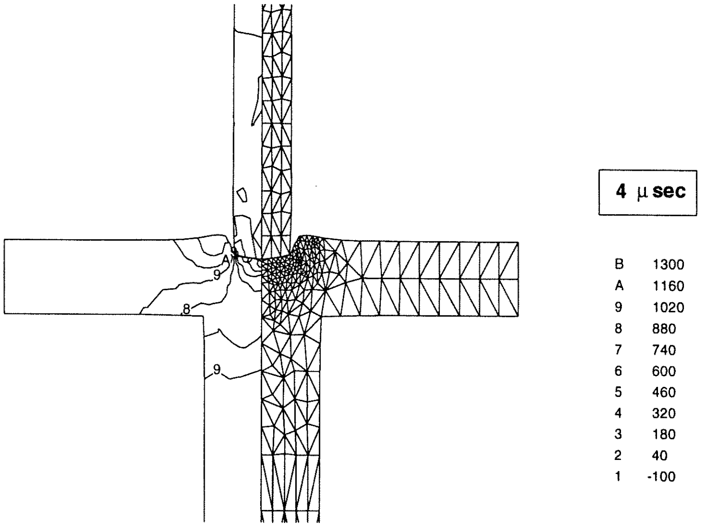

Figure 11: **Axial velocity contours at t=4μs.**

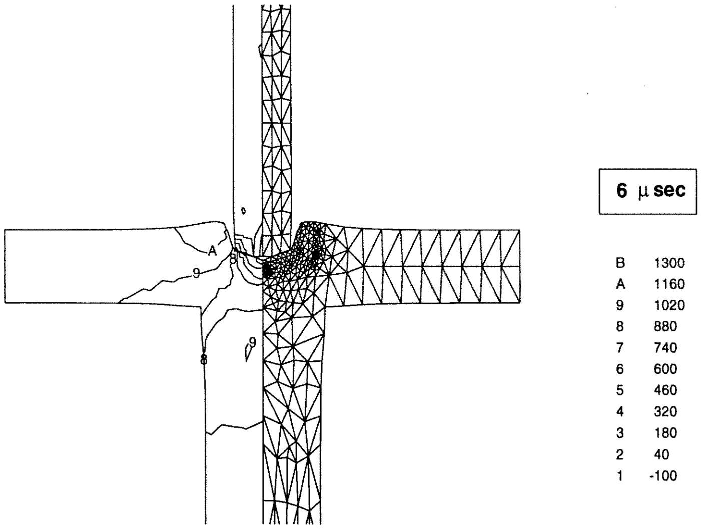

Figure 12: **Axial velocity contours at t=6μs.**

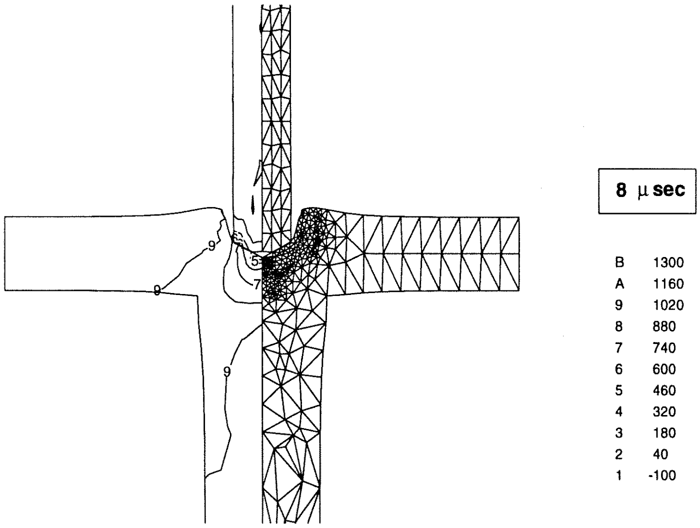

Figure 13: **Axial velocity contours at t=8μs.**

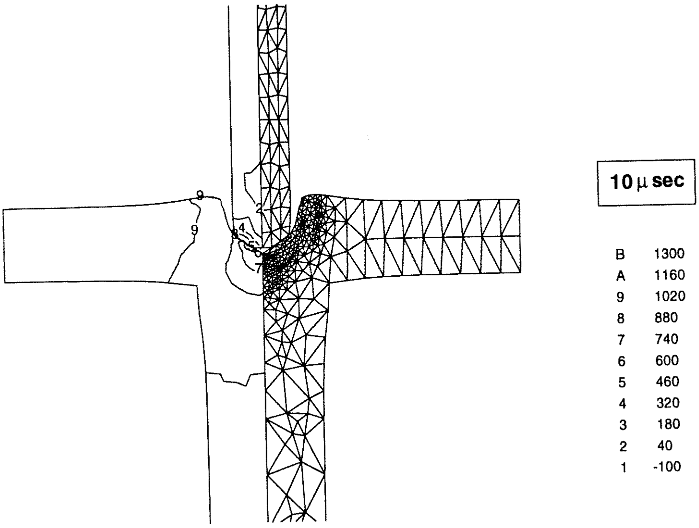

Figure 14: **Axial velocity contours at t=10μs.**

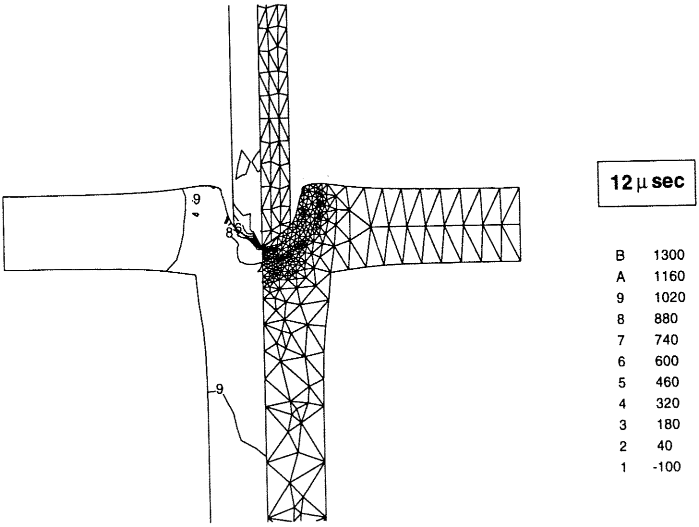

Figure 15: **Axial velocity contours at t=12μs.**

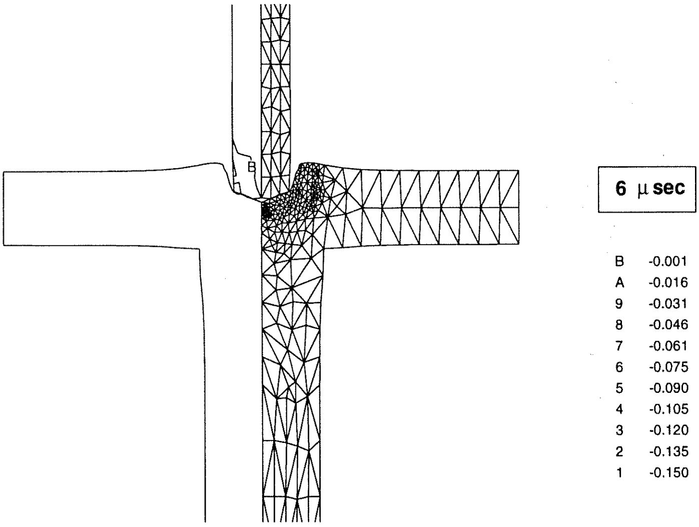

Figure 16: **Contours of second invariant of inelastic strain tensor at t=6μs.**

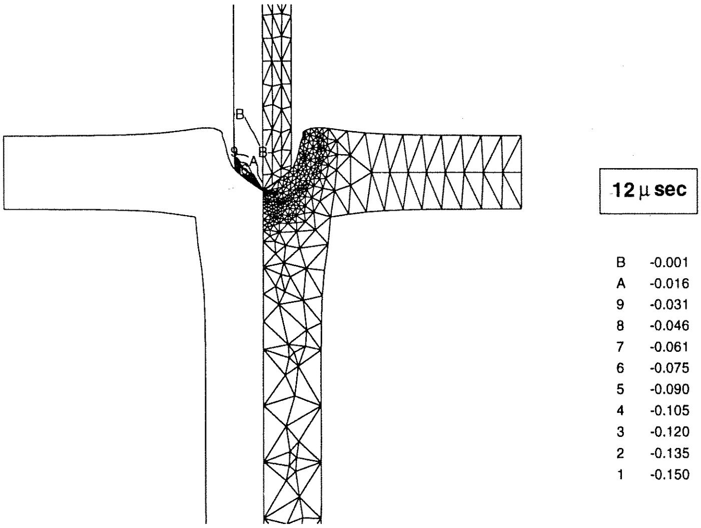

Figure 17: **Contours of second invariant of inelastic strain tensor at t=12μs.**

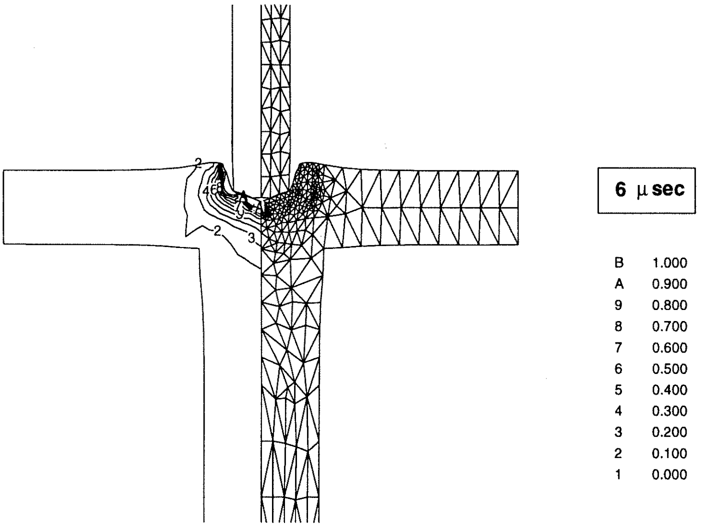

Figure 18: **Effective plastic strain contours at t=6μs.**

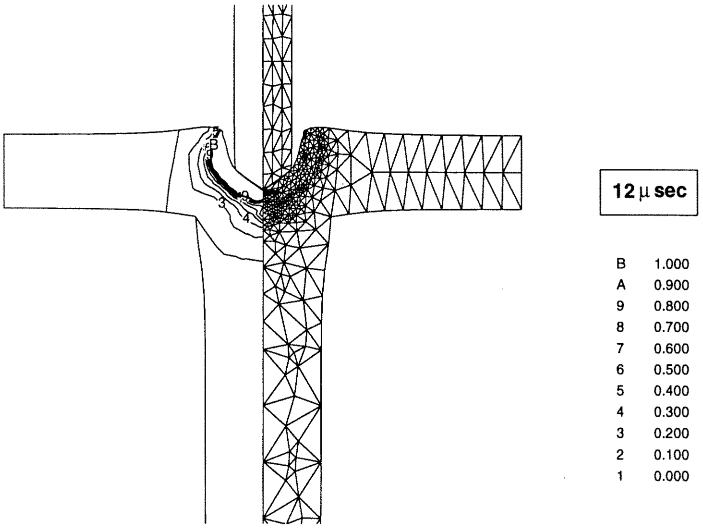

Figure 19: **Effective plastic strain contours at t=12μs.**

rate of plastic work condition mentioned previously. Each time, all the elements with a work rate ratio of $\beta_{max} > 1.1$ were refined. Note that elements that were smaller than $h_{min} = .5$mm across were not refined. One test of the interpolation and recomputation of the state variables due to a mesh rezoning is whether the mesh continues on after the rezoning without any sudden changes in the field variables. Upon examination of the velocity history in Figure 9, it can be seen that no discontinuities in velocity or acceleration are introduced by mesh refinement. One interesting aspect of this problem is that the most rapid plastic deformation in the aluminum, as highlighted by the mesh refinement, is confined to a band of roughly constant width, instead of producing a smooth gradient over a larger area.

In Figures 16 and 17, the contours of the second invariant of the inelastic deformation tensor are depicted as a measure of the degree of microfracture in the alumina ceramic rod. At 6 microseconds, halfway through the calculation, the deformation is primarily elastic, with the inelastic deformation due to cracking confined to one rod diameter. At 12 microseconds, however, the bottom row of elements in the alumina have suffered a tenfold increase in cracking and are about to collapse. Note that most of the inelastic deformation is concentrated in the elements on the very end of the rod and in a spall region near the back which trapped the rebounding tensile wave (not shown). Also displayed, in Figures 18 and 19, are contours of the effective plastic strain in the aluminum rod at 6 and 12 microseconds into the calculation. Note that at 6 microseconds after collision, although the plastic strain is severe right near the surface, it possesses a fairly smooth gradient down to the low-strain regions. At 12 microseconds, however, the edge of the zone of large-scale plastic deformation has become well defined, where the effective plastic strain leaps from around .3 to greater than 1.0 in a very steep gradient. What is interesting is that the contours for the low strain ($< .4$) do not change very much between the two times. Most of the strain localizes in a band near the surface. Consistent with the formation of this band is the accumulation of small elements resulting from the mesh refinement algorithm previously discussed.

ACKNOWLEDGEMENTS

This research was supported by the National Science Foundation through Grants Nos. MSS-9309006, MSS 9311289, CSM-9523113, and by the Army Office of Scientific Research through Grant No. DAAH04-96-1-0142. The first author wishes to thank M. Ortiz and A. Cuitiño for providing valuable insight into the implementation of the large deformation plasticity algorithm.

REFERENCES

Asaro, R.J., and Rice, J.R., 1977, 'Strain localization in ductile single crystals,' *J. Mech. Phys. Solids*, **25**, 309.

Batra, R.C., and Hwang, J., 1993, 'An adaptive mesh refinement technique for two-dimensional shear band problems,' *Comp. Mech.*, **12**, 255.

Batra, R.C., and Ko, K.I., 1992, 'An adaptive mesh refinement technique for the analysis of shear bands in plane strain compression of a thermoviscoplastic solid,' *Comp. Mech.*, **10**, 369.

Bazant, Z.P., and Gambarova, P.G., 1984, 'Crack shear in concrete: crack band microplane model.' *J. Struct. Engng. ASCE*, **110**(9), 2015.

Belytschko, T., 1983, 'An overview of semidiscretization and time integration procedures,' in *Computational Methods for Transient Analysis*, eds. T. Belytschko and T. Hughes, 301.

Benson, D.J., 1992, 'Computational methods in Lagrangian and Eulerian hydrocodes,' *Comput. Methods Appl. Mech. Eng.,* **99**, 235.

Clifton, R.J. and Klopp, R.W., 1985, 'Pressure-Shear Plate Impact Testing', *Metals Handbook: Mechanical Testing 8, 9th Edition*, p. 230.

Donea, J., 1983, 'Arbitrary Lagrangian-Eulerian finite element methods,' in *Computational Methods for Transient Analysis*, eds. T. Belytschko and T. Hughes, 473.

Duffy, J., 1979, 'The dynamic plastic behavior of structural materials: A review,' prepared for the Engineering Foundation.

Dvorkin, E.N., and Petocz, E.G., 1992, 'On the modelling of 2D metal forming processes using flow formulation and the pseudo-concentration technique', in *Proc. 3rd Int. Conf. Comput. Plasticity,* eds. D.R.J. Owen, E. Oñate, and E. Hinton, 1037.

Espinosa, H.D., 1989, 'Finite element analysis of stress induced damage in ceramics,' *MSc. Thesis*, Brown University, Providence, RI.

Espinosa, H.D. and Brar, N.S., 1995, 'Dynamic failure mechanisms of ceramic bars: experiments and numerical simulations,' *J. Mech. Phys. Solids*, **43**(10), 1615.

Espinosa, H.D., Raiser, G., Clifton, R.J., and Ortiz, M., 1992, 'Experimental observations and numerical modeling of inelasticity in dynamically loaded ceramics,' *J. of Hard Materials*, Vol 3, 3-4, pp. 285-313.

Espinosa, H.D., Emore, G., Xu, Y. 1995, 'High Strain rate behavior of composites with continuous fibers', ASME Special Technical Publication, *High Strain Rate Effects on Polymer, Metal and Ceramic Matrix Composites and Other Advanced Materials*, eds. by Y.D.S. Rajapakse and J.R. Vinson, San Francisco, California, 6.

Espinosa, H.D., and Lu, H.C., 1995, Unpublished research.

Flanagan, D.P., and Belytschko, T., 1984, 'Eigenvalues and stable time steps for the uniform strain hexahedron and quadrilateral,' *J. Appl. Mech.,* **51**, 35.

Freund, L.B., 1990, *Dynamic Fracture Mechanics*, Cambridge University Press, Cambridge.

Havner, K.S., 1973, 'On the mechanics of crystalline solids,' *J. Mech. Phys. Solids*, **38**, 87.

Hill, R., and Rice, J.R., 1972, 'Constitutive analysis of elastic-plastic crystals at arbitrary strain,' *J. Mech. Phys. Solids*, **20**, 401.

Ju, J.W., and Lee, X., 1991, 'On three dimensional self-consistent micromechanical damage models for brittle solids,' *J. Engng Mech. ASCE*, **117**(7).

Komanduri, R., 1993, 'Machining and grinding: A historical review of the classical papers,' *Appl Mech Rev*, **46**(3), 80.

Lee, E.H., 1969, 'Elastic plastic deformations at finite strains', *ASME J. Appl. Mech.*, **36**, 1-6.

Liu, W.K., Chang, H., Chen, J.S., and Belytschko, T., 1988, 'Arbitrary Lagrangian-Eulerian Petrov-Galerkin finite elements for nonlinear continua,' *Comput. Methods Appl. Mech. Eng.*, **68**, 259.

Mandel, J., 1972, *Plasticité classique et viscoplasticité*, Course given at Int. Center for Mech. Sci., Udine, Springer, Berlin.

Marusich, T.D., and Ortiz, M., 1995, 'Modelling and simulation of high-speed machining,' *Int J. Num. Meth. Eng.*, **38**, 3675.

Mohan, R., 1991, 'On crack tip fields in ductile single crystals and bicrystals,' *Ph.D. Thesis*, Brown University, Providence, RI.

Needleman, A., 1988, 'Material rate dependence and mesh sensitivity in localization problems,' *Comput. Meth. Engng.*, **67**, 69.

Nicholas, T., and Rajendran, A.M., 1990, 'Material characterization at high strain rates,' in *High velocity impact dynamics*, ed. by J.A. Zukas, John Wiley and Sons, Inc.

Ortiz, M., and Quigley, J.J., 1991, 'Adaptive mesh refinement in strain localization problems,' *Comp. Meth. Appl. Mech. Eng.*, **90**, 781.

Oxley, P.L.B., 1989, *Mechanics of Machining*, Ellis Horwood, Chichester.

Özisik, M.N., 1977, *Basic Heat Transfer*, McGraw-Hill, New York.

Rajendran, A.M., 1992, 'High strain rate behavior of metals, ceramics, and concrete,' *Air Force Report WL-TR-92-4006*, Wright-Patterson Air Force Base, OH.

Rice, J.R., 1971, 'Inelastic constitutive relations for solids: an internal-variable theory and its application to metal plasticity,' *J. Mech. Phys. Solids*, **19**, 433.

Sandler, I.S., and Wright, J.P., 1984, 'Strain-softening,' *Theoretical Foundations for Large Scale Computations of Nonlinear Materials Behavior,* (ed. Nemat Nasser *et al.*, 285.

Seaman, L., and Dein, J.L., 1983, 'Representing shear band damage at high strain rates,' *Proc. IUTAM Symp. on Nonlinear Deformation Waves*,' Tallin, Estonia, Springer-Verlag, Berlin.

Shih, A.J., and Yang, H.T., 1993, 'Experimental and finite element predictions of residual stresses due to orthogonal metal cutting,' *Int J. Num. Meth. Eng.*, **36**, 1487.

Stillman, D.W., Lum, L., Hallquist, J.O., and Benson, D.J., 1993, 'An arbitrary Lagrangian-Eulerian capability for LS-DYNA3D,' *AMD*, **171**, 67.

Taylor, L.M., and Flanagan, D.P., 1987, 'PRONTO 2D: a two dimensional transient solid dynamics program', *Sandia Report SAND86.*

Teodosiu, C., 1970, 'A dynamic theory of dislocations and its applications to the theory of elastic-plastic continuum,' *Fundamental Aspects of Dislocation Theory*, **2**, National Bureau of Standards (U.S.) Special Publication, 837.

Tvergaard, V., 1990, 'Effect of fibre debonding in a whisker-reinforced metal,' *Materials Science and Engineering*, **A125**, 203.

Weber, G., and Anand, L., 1990, 'Finite deformation constitutive equations and a time integration procedure for isotropic, hyperelastic-viscoplastic solids,' *Comp. Meth. Appl. Mech. Eng.*, **79**, 173.

Wise, J.L., and Grady, J.R., 1994, 'Dynamic, multiaxial impact response of confined and unconfined ceramic rods,' in *Proc. High Pressure Science and Technology*, APS Meeting, eds. S.C. Schmidt, J.W. Shaner, G.A. Samara, and M. Ross, 777.

Yang, H.T., Heinstein, M., and Shih, J.M., 1989, 'Adaptive 2D finite element simulation of metal forming processes,' *Int. J. Num. Meth. Eng.*, **28**, 1409.

Zavattieri, P.D., Buscaglia, G.C., and Dari, E.A., 1995, 'Finite element mesh optimization in three dimensions,' Submitted to *Latin American Applied Research.*

Zavattieri, P.D., Dari, E. A., and Buscaglia, G.C., 1995, 'Optimization strategies in unstructured mesh generation,' Submitted to *Int. J. Numer. Meth. in Eng.*.

Zavattieri, P.D., 1995, 'Finite element mesh optimization in three dimensions,' *MSc Thesis in Nuclear Engineering*, Instituto Balseiro, Argentina.

Zhou, M., Ravichandran, G., and Rosakis, A.J., 1996 'Dynamically propagating shear bands in impact-loaded prenotched plates- II. Numerical simulations,' *J. Mech. Phys. Solids*, **44**(6), 1007.

Zhu, J.Z., Himton, E. and Zienkiewicz, O.C., 1991, 'Adaptive finite element analysis with quadrilaterals. *Computers & Structures*, **40**(5), 1097.

Zienkiewicz, O.C., and Taylor, R.L., 1993a, *The Finite Element Method, 4th Ed.*,**1**.

Zienkiewicz, O.C., and Taylor, R.L., 1993b, *The Finite Element Method, 4th Ed.*, **2**.

Zukas, J.A. , 1990, *High velocity impact dynamics* John Wiley and Sons, Inc., New York.