

NOTICE: THIS MATERIAL MAY BE PROTECTED BY COPYRIGHT LAW  
(TITLE 17 U.S. CODE)

# A generalized particle algorithm for high velocity impact computations

G. R. Johnson, S. R. Beissel, R. A. Stryk

**Abstract** This paper presents a Generalized Particle Algorithm (GPA) for high velocity impact and other dynamics problems. A velocity smoothing algorithm is also presented. This generalized algorithm allows for both variable nodal connectivity and fixed nodal connectivity. The variable connectivity option allows for severe distortions with a Lagrangian approach. With fixed nodal connectivity and smoothing it is possible to provide stable computations for large tensile strains. The algorithms are provided for 2D axisymmetric geometry and 3D geometry, and examples are included to demonstrate some of the capabilities. A discussion of interface problems and solutions is also included.

## 1

### Introduction

In recent years there has been an increased emphasis on the development of particle methods for high velocity impact computations. The appeal of particle methods is that they are Lagrangian techniques that can have variable nodal connectivity for severe distortions. The Lagrangian feature is desired because it allows the grid to be embedded in the material and this reduces some of the material interface problems associated with Eulerian codes. Furthermore, the ability to handle severe distortions allows the particle techniques to be applied to problems that historically have been reserved for Eulerian approaches. The Lagrangian formulation also allows the particle algorithms to be linked to Lagrangian finite element algorithms in a straightforward manner.

It is not possible to review all of the previous work concerning particle methods. A recent volume of *Computer Methods in Applied Mechanics and Engineering* is dedicated to recent advances in meshless methods [1] and it provides a comprehensive collection of a variety of particle (or meshless) algorithms. Some of the key historical contributions, most closely associated with the work presented in this paper, are as follows:

In 1977 Lucy [2] and Gingold and Monaghan [3] introduced the Smooth Particle Hydrodynamics (SPH) approach, and two comprehensive SPH reviews are

presented by Benz [4] and Monaghan [5]. A significant contribution to SPH was the addition of strength by Libersky and Petschek in 1990 [6]. More recently, a Normalized Smooth Function (NSF) algorithm was presented by Johnson and Beissel [7]. This algorithm increases the accuracy of the basic SPH algorithm and enables it to pass the patch test, which means that it produces the correct constant strain rates for a linear velocity field.

A completely different particle algorithm, designated NABOR, was presented by Johnson for liquids in 1983 [8], and by Johnson, Stryk, and Dodd for solids in 1986 [9]. The 1986 reference includes the linking together of NABOR (particle) nodes and finite elements. Some recent examples of linking are provided by Johnson, Stryk, and Beissel [10], and by Swegle and Attaway [11].

This paper presents a Generalized Particle Algorithm (GPA). Although it appears to be similar to SPH, it is not SPH because there is no requirement for the smoothing function to have the characteristics of a Dirac delta function. Some of the features of the GPA method are as follows:

- It can be used with variable nodal connectivity to represent severe distortions. The phrase variable nodal connectivity means that a given node can have different groups of neighbor nodes throughout the computation. In this mode it exhibits the positive features of SPH.
- It can be used with fixed nodal connectivity for problems that do not experience severe distortions. The phrase fixed nodal connectivity means that a given node has the same group of neighbor nodes throughout the computation. This requires much less CPU time because each node requires fewer neighbor nodes, as well as no searching for new nearest neighbors. In two dimensions it is possible to use only four fixed neighbors, whereas it is generally necessary to use 8–12 neighbors for variable nodal connectivity. An additional feature of the fixed connectivity option is that it is possible to represent large tensile strains with the use of velocity smoothing.
- It can be used with fixed nodal connectivity at the beginning of the computation, and then the highly distorted portions of the problem can be automatically converted to variable nodal connectivity as the problem progresses. This minimizes the number of nodes that requires more neighbors and more searching, which are associated with variable connectivity. This option can significantly reduce CPU time for some problems.

G. R. Johnson (✉), S. R. Beissel, R. A. Stryk  
Alliant Techsystems Inc., Hopkins, Minnesota, 55343, USA

This work was primarily funded by Contract F08630-92-C-0006 from the Air Force Research Laboratory (AFRL) at Eglin Air Force Base. The authors also appreciate the contributions of W.H. Cook from AFRL.

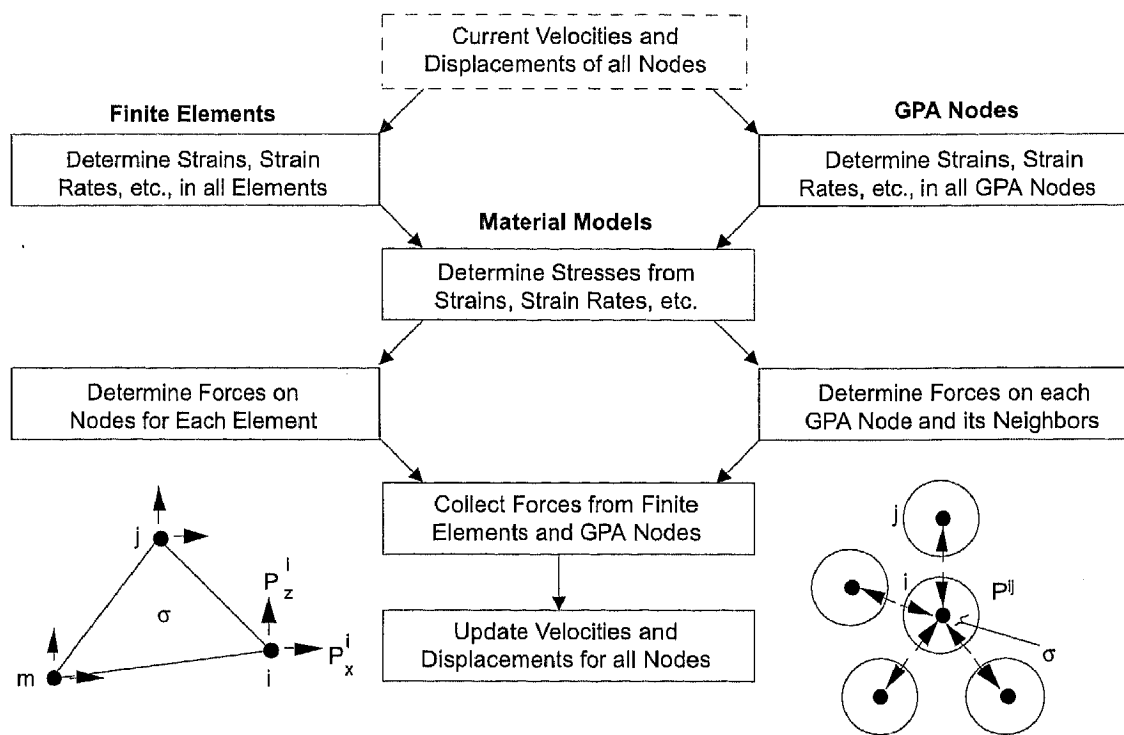


Fig. 1. Lagrangian code structure for finite elements and GPA nodes

This paper presents the 2D axisymmetric GPA method in detail, and it provides a brief 3D algorithm. The nomenclature is very similar to a recent SPH paper by the authors [10]. Although the GPA method has been significantly influenced by the authors experience with SPH, it is clearly not an SPH method (primarily because it does not require the smoothing function to have characteristics of a Dirac delta function).

## 2 GPA method for 2D axisymmetric geometry

A schematic overview of the structure of a Lagrangian code is shown in Fig. 1. It is similar for both finite elements and GPA nodes, with the primary differences being the computations of the strains, strain rates, and nodal forces. The determination of the nodal displacements and velocities, as well as the stresses in the finite elements or GPA nodes, is identical for both approaches and is not included herein. Because both the finite element method and the GPA method are Lagrangian, it is possible to link the two methods together in a straightforward manner [10, 11].

Figure 2 represents some features of the GPA technique. Node  $i$  is designated as the center node and the neighbor nodes are designated as nodes  $j$ . The distance between nodes is  $r_{ij}$ , the diameters of the nodes are  $d_i$  and  $d_j$ , and the masses of the nodes are  $M_i$  and  $M_j$ . The masses remain constant throughout the computation, and are obtained from  $M = \rho_0 V_0$  where  $\rho_0$  and  $V_0$  represent the initial density of the material and the initial volume represented by the node.

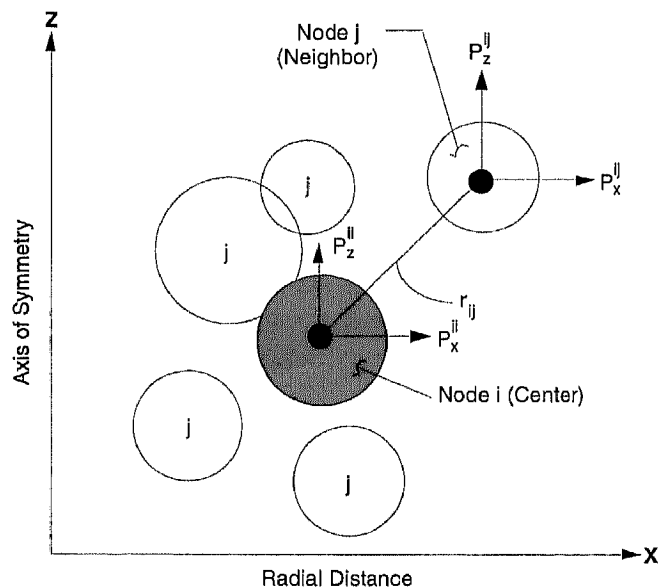


Fig. 2. GPA characteristics

### 2.1 Weighting functions

To determine the strain rates for each center node  $i$ , the contributions of the neighbor nodes  $j$  must be considered. The effect of the neighbor nodes is dependent on the distance between nodes  $i$  and  $j$ , the size (2D area or 3D volume) of node  $j$ , and the alignment of node  $j$  relative to the direction of the strain rate being determined. These same weighting factors are used for the determination of the forces.

For fixed  
function is  
 $f(r) = R_{ij}$   
for this case  
For vari  
tion decre  
 $f(r) = R_{ij}$   
where  $v_{ij} =$   
 $h_{ij} = \alpha(d_i$   
The dimen  
input. It is  
used (0.8  
obtained  
 $d = d_0 \sqrt{1 + \epsilon_v}$   
where  $\epsilon_v$  is  
initial nod  
 $r$  (radial)  
is 2.0 node  
to be cons  
The we  
rectly prop  
by the no  
 $f(s) = A_j$   
where  $V_{ij}$  is  
radial coo  
The we  
with the s  
 $f(\theta_x) = \ell_x^j$   
 $f(\theta_z) = \ell_z^j$   
where  $\theta_x$  is  
and  $j$ , with  
The corre  
2.2  
Strain rate  
For axisym  
 $(\dot{\epsilon}_x, \dot{\epsilon}_z, \dot{\epsilon}_\theta)$   
and the vo  
follows:  
 $\dot{\epsilon}_x = \alpha_x \sum_j$   
 $\dot{\epsilon}_z = \alpha_z \sum_j$   
 $\dot{\epsilon}_\theta = \dot{u}_i / x_i$   
 $\dot{\epsilon}_{xz} = \sum_j$   
 $\dot{\epsilon}_{xz} = \sum_j$

For fixed nodal connectivity the distance weighting function is simply

$$f(r) = R_{ij} \equiv 1.0 \quad (1a)$$

For this case the influence is not affected by distance.

For variable connectivity the distance weighting function decreases with distance.

$$f(r) = R_{ij} = 1.0 - v_{ij}/2 \quad (0 \leq v_{ij} \leq 2) \quad (1b)$$

where  $v_{ij} = r_{ij}/h_{ij}$ , and the influence distance is

$$h_{ij} = \alpha(d_i + d_j)/2 \quad (2)$$

The dimensionless influence distance,  $\alpha$ , is a user supplied input. It is usually taken as  $\alpha = 1.0$ , but other values can be used ( $0.8 \leq \alpha \leq 1.5$ ). The diameters,  $d_i$  and  $d_j$ , can be obtained (for axisymmetric geometry) from

$$d = d_0 \sqrt{(1 + \varepsilon_v)x_0/x} \quad (3)$$

where  $\varepsilon_v$  is the volumetric strain (defined later),  $d_0$  is the initial node diameter, and  $x_0$  and  $x$  are the initial and current  $x$  (radial) coordinates. The radius of influence is  $2h_{ij}$ , which is 2.0 node diameters for  $\alpha = 1.0$ . This definition is used to be consistent with previous SPH formulations.

The weighting function for the size of the node is directly proportional to the cross-sectional area represented by the node.

$$f(s) = A_j = V_j/2\pi x_j \quad (4)$$

where  $V_j$  is the volume represented by node  $j$ , and  $x_j$  is the radial coordinate.

The weighting functions for the alignment of node  $j$  with the strain rates of interest are given by

$$f(\theta_x) = \ell_x^2 \quad (5)$$

$$f(\theta_z) = \ell_z^2 \quad (6)$$

where  $\theta_x$  is the angle between the line connecting nodes  $i$  and  $j$ , with the  $x$  axis, and  $\theta_z$  is the angle with the  $z$  axis. The corresponding direction cosines are  $\ell_x$  and  $\ell_z$ .

## 2.2 Strain rates

For axisymmetric geometry, the three normal strain rates ( $\dot{\varepsilon}_x, \dot{\varepsilon}_z, \dot{\varepsilon}_\theta$ ), the shear strain rate,  $\dot{\gamma}_{xz}$ , the rotational rate,  $\omega_{xz}$ , and the volumetric strain rate,  $\dot{\varepsilon}_v$ , for center node  $i$ , are as follows:

$$\dot{\varepsilon}_x = \alpha_x \sum_j R_{ij} V_j (\dot{u}_j - \dot{u}_i) \ell_x / 2\pi x_j r_{ij} \quad (7)$$

$$\dot{\varepsilon}_z = \alpha_z \sum_j R_{ij} V_j (\dot{v}_j - \dot{v}_i) \ell_z / 2\pi x_j r_{ij} \quad (8)$$

$$\dot{\varepsilon}_\theta = \dot{u}_i / x_i \quad (9)$$

$$\dot{\gamma}_{xz} = \sum_j R_{ij} V_j [\alpha_z (\dot{u}_j - \dot{u}_i) \ell_z + \alpha_x (\dot{v}_j - \dot{v}_i) \ell_x] / 2\pi x_j r_{ij} \quad (10)$$

$$\omega_{xz} = \sum_j R_{ij} V_j [\alpha_x (\dot{v}_j - \dot{v}_i) \ell_x - \alpha_z (\dot{u}_j - \dot{u}_i) \ell_z] / 4\pi x_j r_{ij} \quad (11)$$

$$\dot{\varepsilon}_v = \dot{\varepsilon}_x + \dot{\varepsilon}_z + \dot{\varepsilon}_\theta \quad (12)$$

where  $R_{ij}$  is the distance weighting function (for both fixed and variable nodal connectivity),  $V_j$  is the current volume of the node  $j$ ,  $\dot{u}_i$  and  $\dot{u}_j$  are the  $x$  velocities of nodes  $i$  and  $j$ ,  $\dot{v}_i$  and  $\dot{v}_j$  are the  $z$  velocities,  $\ell_x$  and  $\ell_z$  are the direction cosines from node  $i$  to node  $j$ ,  $x_j$  is the  $x$  coordinate of node  $j$ , and  $r_{ij}$  is the distance between nodes  $i$  and  $j$ . The volumetric strain is obtained by integrating the volumetric strain rate.

$$\varepsilon_v^{t+\Delta t} = \varepsilon_v^t + \dot{\varepsilon}_v \Delta t (1 + \varepsilon_v^t) \quad (13)$$

where  $\Delta t$  is the integration time increment and the factor  $(1 + \varepsilon_v^t)$  converts the incremental strain ( $\dot{\varepsilon}_v \Delta t$ ) from the current configuration back to the initial configuration. The volumetric strain,  $\varepsilon_v = V/V_0 - 1$ , must be based on the initial volume for the pressure computations, where  $V$  is the current volume and  $V_0$  is the initial volume. Because  $\dot{\varepsilon}_v$  in Eq. (12) is based on the current configuration, it must be converted back to the initial configuration.

The remaining two geometry factors in Eqs. (7)–(11) are

$$\alpha_x = \frac{1}{\sum_j R_{ij} V_j \ell_x^2 / 2\pi x_j} \quad (14)$$

$$\alpha_z = \frac{1}{\sum_j R_{ij} V_j \ell_z^2 / 2\pi x_j} \quad (15)$$

It should be noted that these geometry factors are obtained in the global coordinate system, rather than a local (invariant) coordinate system about each individual node. For interior nodes this is not generally important, but for boundary and interface nodes it can be significant. This effect is addressed in more detail in the discussion of interface effects.

The derivation of the preceding equations begins with the strain rate in the  $x$  direction.

$$\dot{\varepsilon}_x = \frac{\partial \dot{u}}{\partial x} = \frac{\sum_j \left( \frac{\Delta \dot{u}_{ij}}{\Delta x_{ij}} \right) f(r) f(s) f(\theta_x)}{\sum_j f(r) f(s) f(\theta_x)} \quad (16)$$

where  $\Delta \dot{u}_{ij} = \dot{u}_j - \dot{u}_i$  is the  $x$  velocity difference between nodes  $i$  and  $j$ , and  $\Delta x_{ij}$  is the  $x$  coordinate difference. Basically, all values of  $\Delta \dot{u}_{ij} / \Delta x_{ij}$  are weighted by the distance function,  $f(r)$ , the size function,  $f(s)$ , and the alignment function with the  $x$  axis,  $f(\theta_x)$ . It is clear that the strain rate in Eq. (16) passes the patch test because  $\dot{\varepsilon}_x \equiv \Delta \dot{u}_{ij} / \Delta x_{ij}$  for a linear velocity field (where  $\Delta \dot{u}_{ij} / \Delta x_{ij}$  is constant).

If the weighting functions of Eqs. (1), (4), and (5) are substituted into Eq. (16), the  $x$  strain rate is expressed as

$$\dot{\varepsilon}_x = \frac{\partial \dot{u}}{\partial x} = \frac{\sum_j \left( \frac{\Delta \dot{u}_{ij}}{\Delta x_{ij}} \right) R_{ij} V_j \ell_x^2 / 2\pi x_j}{\sum_j R_{ij} V_j \ell_x^2 / 2\pi x_j} \quad (17)$$

Setting  $\ell_x^2 / \Delta x_{ij} = \ell_x / r_{ij}$ , and substituting  $\alpha_x$  from Eq. (14) into Eq. (17), gives the form of Eq. (7).

The strain rate in the  $z$  direction is obtained in the same manner.

$$\dot{\epsilon}_z = \frac{\partial \dot{v}}{\partial z} = \frac{\sum_j \left( \frac{\Delta v_{ij}}{\Delta z_{ij}} \right) R_{ij} V_j \ell_z^2 / 2\pi x_j}{\sum_j R_{ij} V_j \ell_z^2 / 2\pi x_j} \quad (18)$$

where  $\Delta v_{ij} = \dot{v}_j - \dot{v}_i$  is the  $z$  velocity difference and  $\Delta z_{ij}$  is the  $z$  coordinate difference. Again, setting  $\ell_z^2 / \Delta z_{ij} = \ell_z / r_{ij}$ , and substituting  $\alpha_z$  from Eq. (15) into Eq. (18), gives the form of Eq. (8).

The hoop strain rate in Eq. (9) is determined only from the  $x$  velocity and  $x$  coordinate of node  $i$ . The neighbor nodes  $j$  are not used. Finally, the shear strain rate and rotational rate in Eqs. (10) and (11) are obtained from

$$\dot{\gamma}_{xz} = \frac{\partial \dot{u}}{\partial z} + \frac{\partial \dot{v}}{\partial x} \quad (19)$$

$$\omega_{xz} = \frac{1}{2} \left( \frac{\partial \dot{v}}{\partial x} - \frac{\partial \dot{u}}{\partial z} \right) \quad (20)$$

The procedures are similar to those used in Eqs. (16)–(18).

### 2.3

#### Forces

After the strain rates, rotational rate, and volumetric strain in Eqs. (7)–(13) are determined, it is possible to determine the shear and deviator stresses, the pressure, and the nodal artificial viscosity in the standard manner [12]. These stresses must then be converted to forces as shown in Fig. 1.

The final forms of the force equations are as follows:

$$P_x^{ij} = \frac{-R_{ij} V_i V_j}{2\pi x_j r_{ij}} [\alpha_x (\sigma_x^i - Q_{ij}) \ell_x + \alpha_z \tau_{xz}^i \ell_z] \quad (21)$$

$$P_z^{ij} = \frac{-R_{ij} V_i V_j}{2\pi x_j r_{ij}} [\alpha_z (\sigma_z^i - Q_{ij}) \ell_z + \alpha_x \tau_{xz}^i \ell_x] \quad (22)$$

$$P_x^{ii} = - \sum_j P_x^{ij} - \sigma_\theta V_i / x_i \quad (23)$$

$$P_z^{ii} = - \sum_j P_z^{ij} \quad (24)$$

In Eq. (21)  $P_x^{ij}$  is the force on node  $j$  due to the stresses in node  $i$ . The normal stress in the  $x$  direction (for node  $i$ ) is  $\sigma_x^i = s_x^i - (P_i + Q_i)$ , where  $s_x^i$  is the deviator stress,  $P_i$  is the hydrostatic pressure and  $Q_i$  is the nodal artificial viscosity. The shear stress is  $\tau_{xz}^i$ . There is also an artificial viscosity,  $Q_{ij}$ , which is dependent on the relative velocities of nodes  $i$  and  $j$  [13]. It is intended to stabilize the grid and keep adjacent nodes from becoming too close to one another. This is designated as a bond viscosity, because it acts on the bond between nodes  $i$  and  $j$ . It can also introduce a significant amount of artificial strength [14].

The derivation of the preceding force equations begins with the plane strain equilibrium equation in the  $x$  direction.

$$\frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xz}}{\partial z} = \rho \ddot{u} \quad (25)$$

where  $\sigma_x$  and  $\tau_{xz}$  are the normal and shear stresses,  $\rho$  is the density and  $\ddot{u}$  is the acceleration in the  $x$  direction. The density of node  $i$  can be expressed as  $\rho_i = M_i / V_i$ , where

$M_i$  and  $V_i$  are the mass and volume of node  $i$ . Also, the acceleration of node  $i$  can be expressed as  $\ddot{u}_i = P_x^i / M_i$ , where  $P_x^i$  is the net force (in the  $x$  direction) acting on node  $i$ . Substituting for  $\rho_i$  and  $\ddot{u}_i$  in Eq. (25) leads to

$$P_x^i = V_i \left( \frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xz}}{\partial z} \right) \quad (26)$$

Now, if the terms  $\partial \sigma_x / \partial x$  and  $\partial \tau_{xz} / \partial z$  are determined in the same manner as the strain rate in the  $x$  direction (from Eqs. (7), (16) and (17)), then Eq. (26) becomes

$$P_x^i = V_i \left[ \alpha_x \sum_j \left( \frac{\Delta \sigma_x}{\Delta x_{ij}} \right) R_{ij} A_j \ell_x^2 + \alpha_z \sum_j \left( \frac{\Delta \tau_{xz}}{\Delta z_{ij}} \right) R_{ij} A_j \ell_z^2 \right] \quad (27)$$

where most of the terms are equal to, or analogous to, those in the strain rate equations. Additional substitutions of  $\Delta \sigma_x = \sigma_x^j - \sigma_x^i$ ,  $\Delta \tau_{xz} = \tau_{xz}^j - \tau_{xz}^i$ ,  $\Delta x_{ij} = r_{ij} \ell_x$ , and  $\Delta z_{ij} = r_{ij} \ell_z$ , give

$$P_x^i = V_i \left[ \alpha_x \sum_j \frac{(\sigma_x^j - \sigma_x^i)}{r_{ij}} R_{ij} A_j \ell_x + \alpha_z \sum_j \frac{(\tau_{xz}^j - \tau_{xz}^i)}{r_{ij}} R_{ij} A_j \ell_z \right] \quad (28)$$

It should be recalled that Eq. (28) is the equilibrium equation in the  $x$  direction for plane strain geometry. Furthermore, the force is dependent on stresses from node  $i$ , as well as from neighbor nodes  $j$ .

Using the form of Eq. (28) has some disadvantages. The first is that it does not ensure equal and opposite forces between the center node  $i$  and the neighbor nodes  $j$ , and this can lead to changes in the momentum of the system. Secondly, it requires the stresses for all of the nodes to be determined before the forces can be determined. This can cause some computational inefficiencies. Thirdly, it can introduce significant errors on boundary nodes.

The algorithm in this paper uses an alternate approach that considers the forces on nodes  $i$  and  $j$ , due to the stresses in node  $i$  only. These forces are equal and opposite such that momentum is absolutely conserved. The stresses in neighbor node  $j$  are considered when node  $j$  becomes center node  $i$ , and center node  $i$  becomes neighbor node  $j$ . Furthermore, with this approach it is possible to compute some of the forces before all of the stresses are determined. As will be shown later, this approach can also introduce errors on boundary nodes.

If the basic form of Eq. (28) is used to represent the relationship between stresses and forces, then the force on node  $j$ , due to the stress in node  $i$ , can be expressed as

$$P_x^{ij} = -V_i [\alpha_x \sigma_x^i R_{ij} A_j \ell_x / r_{ij} + \alpha_z \tau_{xz}^i R_{ij} A_j \ell_z / r_{ij}] \quad (29)$$

It can also be shown that this alternate approach is identical to Eq. (28), if  $\alpha_x$  at node  $i$  is equal to  $\alpha_x$  at nodes  $j$ . Equation 29 can be rearranged to give

$$P_x^{ij} = \frac{-R_{ij} V_i A_j}{r_{ij}} [\alpha_x \sigma_x^i \ell_x + \alpha_z \tau_{xz}^i \ell_z] \quad (30)$$

isymmetric ge  
 $V_j / 2\pi x_j$ . Also, i  
isymmetric ge  
 $\frac{-R_{ij} V_i V_j}{2\pi x_j r_{ij}} [\alpha_x (\sigma_x^i - Q_{ij}) \ell_x + \alpha_z \tau_{xz}^i \ell_z]$   
is identical to  
the force on nod  
follows:  
 $- \sum_j P_x^{ij} - \sigma_\theta V_i / x_i$   
first term is sim  
on the  $j$  node  
stress,  $\sigma_\theta$ .  
The effect of the  
ing the  $x$  comp  
lar segment  $\Delta$   
hoop) =  $-\sigma_\theta^i V_i / x_i$   
cause this force  
 $(1/2\pi) M_i$ , it is ne  
that the total ho  
as,  $M_i$ . This mul  
 $= V_i / 2\pi x_i$ , gives  
hoop) =  $-\sigma_\theta^i V_i / x_i$   
stituting Eq. (3  
 $- \sum_j P_x^{ij} - \sigma_\theta V_i / x_i$   
which is identical  
It should be not  
de  $j$  has not bee  
p. (29)–(35). Th  
comes center n  
neighbor node  $j$ .  
The forces in th  
forces in the  $z$   
res in the  $z$  dir  
be derived in  
res in the  $x$  dir  
Although it has  
Eq. (29) has so  
advantages for th  
p. (28). The aut  
valuation of the  
object of future  
Artificial viscosity  
There are two for  
cosity,  $Q_i$ , and th  
distinction betwe  
an only be activ  
uch as pure she  
that  $Q_{ij}$  can int  
be computed re  
The nodal vis  
ement and fini

For axisymmetric geometry the area becomes

$A_j = V_j/2\pi x_j$ . Also, if the bond viscosity is added, then  $P_x^{ij}$  for axisymmetric geometry becomes

$$P_x^{ij} = \frac{-R_{ij}V_iV_j}{2\pi x_j r_{ij}} [\alpha_x(\sigma_x^i - Q_{ij})\ell_x + \alpha_z\tau_{xz}^i\ell_z] \quad (31)$$

which is identical to Eq. (21).

The force on node  $i$ , due to the stresses in node  $i$ , is as follows:

$$P_x^i = - \sum_j P_x^{ij} - P_{x(\text{hoop})}^i \quad (32)$$

The first term is simply a sum of the equal and opposite forces on the  $j$  nodes, and the second term is due to the hoop stress,  $\sigma_\theta$ .

The effect of the hoop stress can be obtained by considering the  $x$  component of the hoop stress for a small angular segment  $\Delta\phi$ .

$$\Delta P_{x(\text{hoop})}^i = -\sigma_\theta A_i \Delta\phi \quad (33)$$

Because this force acts on a segment of mass that is  $(\Delta\phi/2\pi)M_i$ , it is necessary to multiply Eq. (33) by  $2\pi/\Delta\phi$  so that the total hoop force is consistent with the total mass,  $M_i$ . This multiplication, together with setting  $A_i = V_i/2\pi x_i$ , gives

$$P_{x(\text{hoop})}^i = -\sigma_\theta^i V_i/x_i \quad (34)$$

Substituting Eq. (34) into Eq. (32) gives

$$P_x^i = - \sum_j P_x^{ij} - \sigma_\theta V_i/x_i \quad (35)$$

which is identical to Eq. (23).

It should be noted again that the effect of the stresses in node  $j$  has not been included in the forces on node  $i$ , for Eqs. (29)–(35). This effect is included when neighbor node  $j$  becomes center node  $i$ , and center node  $i$  becomes neighbor node  $j$ .

The forces in the  $x$  direction are more complicated than the forces in the  $z$  direction because of the hoop stress. The forces in the  $z$  direction, provided in Eqs. (22) and (24), can be derived in a manner similar to that used for the forces in the  $x$  direction.

Although it has been stated that the alternate approach of Eq. (29) has some advantages, there may also be some advantages for the stress difference formulation of Eq. (28). The authors have not performed a comparative evaluation of the two approaches, and this should be the subject of future work.

## 2.4

### Artificial viscosity

There are two forms of artificial viscosity; the nodal viscosity,  $Q_i$ , and the bond viscosity,  $Q_{ij}$ . The important distinction between  $Q_i$  (nodal) and  $Q_{ij}$  (bond) is that  $Q_i$  can only be activated with volumetric strain rate, but  $Q_{ij}$  can be activated even if there is no volumetric strain rate (such as pure shear or incompressible flow). The net effect is that  $Q_{ij}$  can introduce additional (artificial) strength into the computed results.

The nodal viscosity is identical to that used in finite element and finite difference methods [15, 16]:

$$Q_i = C_L \rho_i c_i h_i |\dot{\epsilon}_v| + C_Q \rho_i h_i^2 \dot{\epsilon}_v^2 \quad (36)$$

for  $\dot{\epsilon}_v < 0$ .  $C_L$  and  $C_Q$  are the linear and quadratic coefficients,  $\rho_i$  is the density of node  $i$ ,  $c_i$  is the sound velocity of node  $i$ , and  $h_i$  is the minimum influence distance between node  $i$  and neighbor nodes  $j$ .

The bond viscosity of Monaghan and Gingold [13] can be expressed in the following form:

$$Q_{ij} = C_L \rho_i c_i |\mu_{ij}| + C_Q \rho_i \mu_{ij}^2 \quad (37)$$

for  $\mu_{ij} < 0$ , and

$$\mu_{ij} = \frac{h_{ij}(\dot{u}_{ij}x_{ij} + \dot{v}_{ij}z_{ij})}{r_{ij}^2 + \epsilon_0 h_{ij}^2} \quad (38)$$

where  $\dot{u}_{ij} = \dot{u}_i - \dot{u}_j$  is the  $x$  velocity difference between nodes  $i$  and  $j$ , and  $x_{ij} = x_i - x_j$  is the  $x$  coordinate difference. The analogous terms for the  $z$  components are  $\dot{v}_{ij}$  and  $z_{ij}$ . The distance between nodes is  $r_{ij} = \sqrt{x_{ij}^2 + z_{ij}^2}$ , and  $\epsilon_0$  is a small number ( $\epsilon_0 \approx 0.01$ ) that acts to limit  $\mu_{ij}$  as  $r_{ij}$  becomes small.

Equation 37 can be rewritten in the following form (for  $\epsilon_0 = 0$ ):

$$Q_{ij} = C_L \rho_i c_i h_{ij} |\dot{\epsilon}_{ij}| + C_Q \rho_i h_{ij}^2 \dot{\epsilon}_{ij}^2 \quad (39)$$

for  $\dot{\epsilon}_{ij} < 0$ . This looks very similar to the nodal artificial viscosity Eq. (36), except the volumetric strain rate,  $\dot{\epsilon}_v$ , of Eq. (36) is replaced with a linear strain rate,  $\dot{\epsilon}_{ij}$ , in Eq. (39). This linear strain rate is simply the strain rate along the bond between nodes:

$$\dot{\epsilon}_{ij} = (V_N^j - V_N^i)/r_{ij} \quad (40)$$

where  $V_N^i$  is the velocity of node  $i$  along the bond from node  $i$  to node  $j$ , and  $V_N^j$  is the velocity of node  $j$  in the same direction. A similar form of this bond viscosity was developed previously for the NABOR particle method algorithm [8, 9].

## 2.5

### Interface effects

The primary thrust of this paper is to present the GPA formulation. There are, however, some errors introduced at material interfaces, and a discussion of the source of these errors is included. There are also some errors on free boundaries, but the stresses are generally lower on free boundaries, and these errors are not as critical. The authors are currently developing algorithms to minimize the errors on both interfaces and boundaries. This work is in progress and will be reported in the future.

The GPA method is very robust and will compute strain rates and forces for interface conditions where some of the neighbor nodes  $j$  do not represent the same material as center node  $i$ . Referring to Fig. 2, if some of the neighbor nodes  $j$  represent a material with much lower strength and stiffness than center node  $i$ , this weaker material will probably experience greater deformations under a given stress field. This will, in turn, affect the strain rates in center node  $i$ , and it will probably tend to overpredict the true strain rates in node  $i$ . It is difficult to generalize the magnitude of these errors as they are problem dependent. Generally, however, the strain rate errors increase as the

strength and stiffness differences between the interface nodes increase.

The obvious solution for this condition is to consider only those neighbor nodes  $j$  that represent the same material as center node  $i$ , for the summations in the strain rate equations. With this approach, however, it is necessary to have special contact/sliding algorithms to accurately represent the interaction of GPA nodes of different materials. In a related manner, for linked computations it is also necessary to have special contact/sliding algorithms to accurately represent the interaction of GPA nodes with adjacent finite elements.

For the remainder of this interface discussion it will be assumed that the neighbor nodes  $j$  represent the same material as center node  $i$ . For this condition the previously presented strain rate equations provide an accurate representation of the strain rates, even at boundaries and interfaces.

Unfortunately, the force equations contain some inaccuracies at boundaries and interfaces. Figures 3 and 4 provide a detailed look at the force distributions on four different nodes; interior node A, interface node B, side node C, and corner interface node D. With an applied pressure of  $P_0$  on the top surface of the nodes, the quasi-static stress state at all nodes is assumed to be  $\sigma_z = -P_0$ . All other stresses are assumed to be zero.

For plane strain geometry with a unit thickness, if the applied pressure is taken as  $P_0 = 1.0$ , and the nodal diameters are all  $d = 1.0$ , then every node should have a net downward force of 1.0 from the nodes above, as well as a net upward force of 1.0 from the nodes below. For this analysis the influence distance in Eq. (2) is  $\alpha = 1.0$ , and this means that the distance weighting function in Eq. (1b) goes to zero at a distance of  $2d$ . The result is that an interior node exchanges forces with the three nodes above it and the three nodes below it.

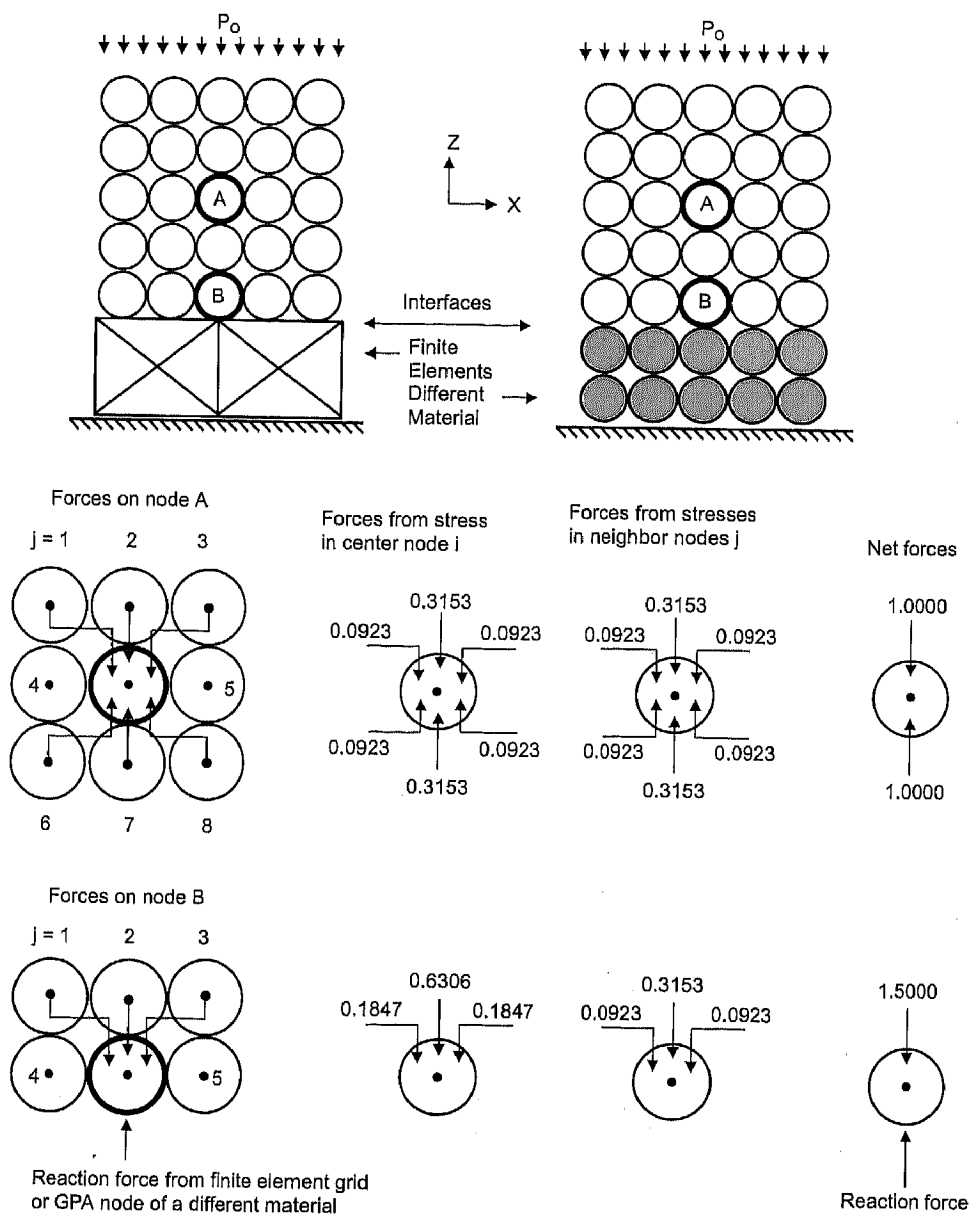
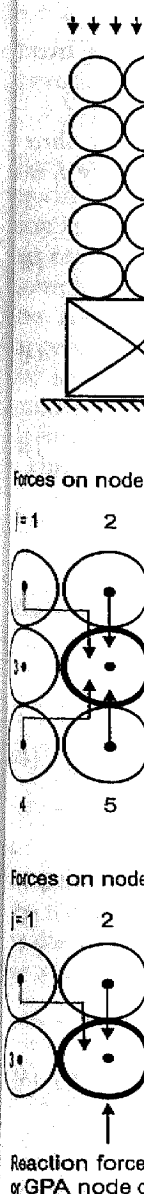


Fig. 3. Forces on interior node A and interface node B



The first ca  
 geometry fact  
 $\alpha_z = 0.6306$ .  
 to substitute  
 all  $A_j = d^2 =$   
 Force Equa  
 strain geomet  
 $P_i^j = -R_{ij}A_i\alpha_z$   
 $P_i^j = R_{ij}\alpha_z\ell_z/$   
 for  $A_i = A_j =$   
 Recall that  
 node  $i$ . For th  
 on node  $j = 1$   
 $\alpha_z = 0.6306$ ,  
 Node  $j = 2$  re

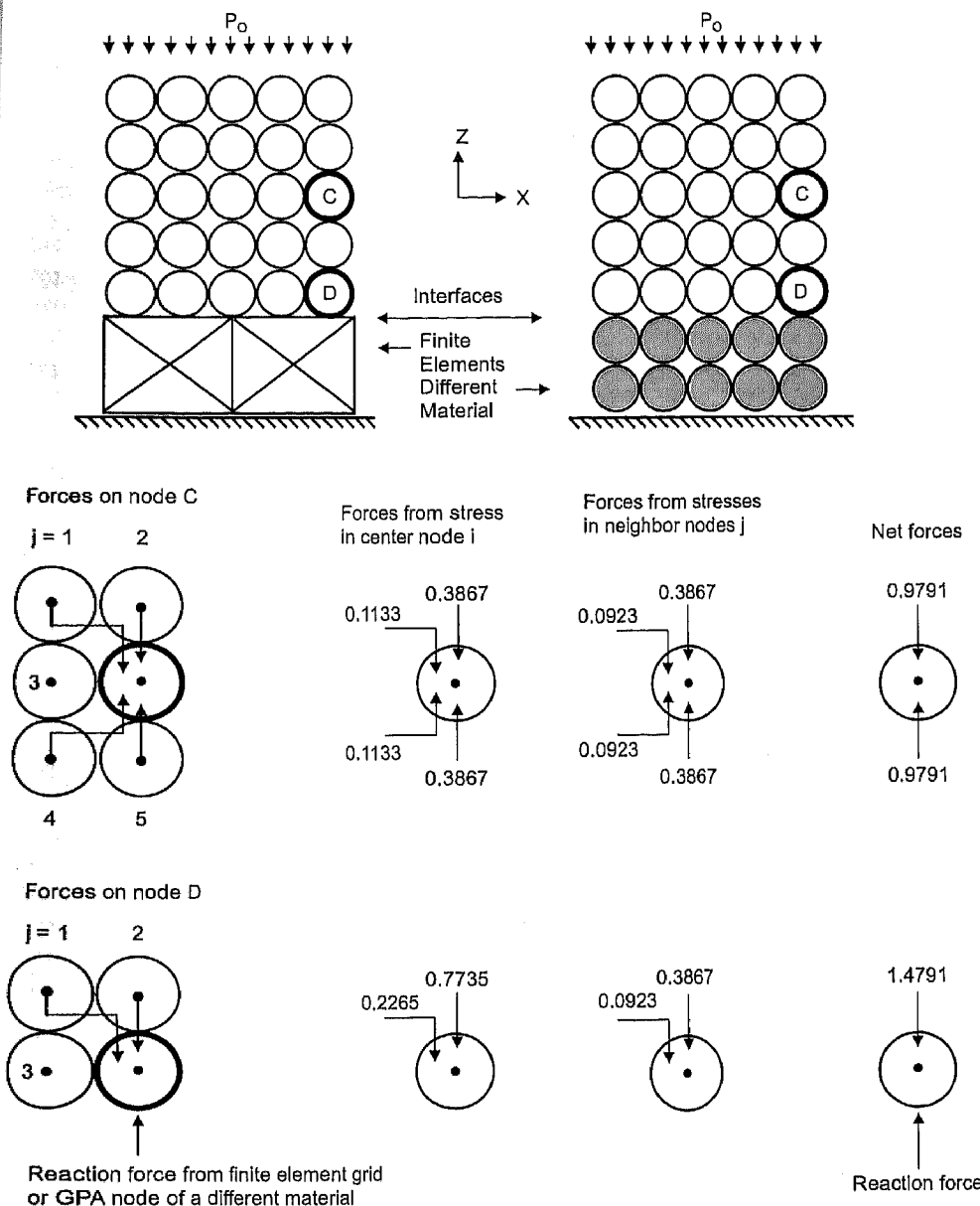


Fig. 4. Force on edge node C and corner interface node D

The first case is interior node A in Fig. 3. To begin, the geometry factor from Eq. (15) is computed to be  $\alpha_z = 0.6306$ . Because this is for plane strain it is necessary to substitute  $A_j$  for  $V_j/2\pi x_j$  in Eq. (15). If all  $d = 1.0$ , then all  $A_j = d^2 = 1.0$ .

Force Equation 22 simplifies to the following for plane strain geometry with no artificial viscosity and no shear:

$$P_z^{ij} = -R_{ij}A_iA_j\alpha_z\sigma_z^i\ell_z/r_{ij} \quad (41a)$$

$$P_z^{ij} = R_{ij}\alpha_z\ell_z/r_{ij} \quad (41b)$$

for  $A_i = A_j = 1.0$  and  $\sigma_z^i = -1.0$ .

Recall that  $P_z^{ij}$  is the force on node  $j$  due to the stress in node  $i$ . For the conditions presently considered, the force on node  $j = 1$ , due to a stress in node  $i$  ( $\sigma_z^i = -1.0$ ), is  $P_z^{ij} = 0.0923$ . This comes from substituting  $R_{ij} = 0.2929$ ,  $\alpha_z = 0.6306$ ,  $\ell_z = 0.7071$  and  $r_{ij} = 1.4142$ , into Eq. (41b). Node  $j = 2$  receives a greater force ( $P_z^{ij} = 0.3153$ ) because

it is closer and more aligned with the  $z$  axis ( $R_{ij} = 0.5000$ ,  $\alpha_z = 0.6306$ ,  $\ell_z = 1.0000$ , and  $r_{ij} = 1.0000$ ). The forces on neighbor nodes  $j$ , due to stresses in center node  $i$ , also act on center node  $i$ , in an equal and opposite manner, as provided in Eq. (24). These are shown under the heading of "Forces from stress in center node  $i$ ," for node A in Fig. 3. Also shown are the "Forces from stresses in neighbor nodes  $j$ ," and these are identical because all of the neighbor nodes are also interior nodes with  $\alpha_z = 0.6306$ . The net forces on node A are exactly 1.0000 acting downward from the three neighbor nodes above ( $j = 1, 2, 3$ ), and exactly 1.0000 acting upward from the three neighbor nodes below ( $j = 6, 7, 8$ ). This tends to confirm the force equations for an interior node.

The forces acting on node B are shown in the lower portion of Fig. 3. This is an interface node that could be in contact with a finite element grid or a GPA node of a different material. A significant difference, between



interface node  $B$  and interior node  $A$ , is that the geometry factor from Eq. (15) is computed to be  $\alpha_z = 1.2612$ . This is exactly twice the magnitude of  $\alpha_z$  for an interior node, and it produces forces from the stress in node  $i$  that are twice those obtained from an interior node. The forces from stresses in neighbor nodes  $j$  are identical to those obtained for interior node  $A$  as the neighbors to node  $B$  ( $j = 1, 2, 3$ ) are all interior nodes with  $\alpha_z = 0.6306$ .

Summing the forces between node  $B$  and neighbor nodes  $j = 1, 2, 3$  gives a net downward force of 1.5000 on node  $B$ , instead of the correct force of 1.0000. The source of this error is that the forces from the stress in node  $i$  are two times higher than they should be, and this is because  $\alpha_z$  is two times higher than for an interior node. Even though the downward force acting on node  $B$  is too high, the system is in equilibrium because every force on neighbor node  $j$  (from the stress in center node  $i$ ) is reacted by an equal and opposite force acting on node  $i$ .

A solution for this interface error is to adjust the values of geometry factors  $\alpha_x$  and/or  $\alpha_z$  in Eq. (14) and/or Eq. (15), for all interface nodes. The adjusted values should be one half of the computed values in the direction normal to the interface.

It was noted previously that the stress difference formulation of Eq. (28) also had some disadvantages for interface nodes. If this formulation would be applied to interface node  $B$ , there would be no net force acting on  $B$  from the neighbor nodes  $j$  because the stress difference would always be  $\sigma_z^j - \sigma_z^i \equiv 0$  (for  $\sigma_z^j = \sigma_z^i$ ). This is a significant discrepancy that would require a specialized boundary algorithm.

Figure 4 shows similar analyses for side node  $C$  and corner interface node  $D$ . For side node  $C$ ,  $\alpha_z = 0.7735$  from Eq. (15), and the net forces acting on both the top and bottom of node  $C$  are 0.9791. This is an error of only 2.1 percent, and the forces tend to cancel one another for a constant stress state. It is only for stress gradients that errors are introduced into the solution, and these errors are small.

The final condition is corner interface node  $D$ , which has  $\alpha_z = 1.5469$  from Eq. (15). Note that  $\alpha_z$  in this case is twice that of  $\alpha_z$  for edge node  $C$ . The net forces acting on node  $D$  from neighbor nodes  $j = 1, 2$  is 1.4791, which is too high. Again, however, if  $\alpha_z$  for node  $D$  is reduced to one half of its computed value, because it is an interface node, then the net force is reduced to 0.9791, which is very close to the correct value of 1.0000.

An important characteristic of the examples in Figs. 3 and 4 is that the boundaries and interfaces are aligned with the global system axes  $x$  and  $z$ , and this affects the determination of the geometry factors ( $\alpha_x$  and  $\alpha_z$ ) in Eqs. (14) and (15). If the boundaries and interfaces are not aligned with the global systems axes then the errors are increased and the correction factors become more complex. The authors are currently working in the development of a frame invariant algorithm that computes  $\alpha_{x'}$  and  $\alpha_{z'}$  along local  $x'$  and  $z'$  axes that are aligned with the boundaries and interfaces. With this alignment the boundary and interface adjustments can be made in a straightforward manner. Another invariant normalization is provided in the tensor formulation of Randles and Libersky [17].

### 3

#### GPA method for 3D geometry

There is very little additional complexity in going from 2D to 3D geometry. Using the same procedures as shown for the 2D geometry, the three normal strain rates ( $\dot{\epsilon}_x, \dot{\epsilon}_y, \dot{\epsilon}_z$ ), the three shear strain rates ( $\dot{\gamma}_{xy}, \dot{\gamma}_{xz}, \dot{\gamma}_{yz}$ ), and the three rotational rates ( $\omega_{xy}, \omega_{xz}, \omega_{yz}$ ) for the center node  $i$ , are as follows:

$$\dot{\epsilon}_x = \frac{\partial \dot{u}}{\partial x} = \alpha_x \sum_j R_{ij} V_j (\dot{u}_j - \dot{u}_i) \ell_x / r_{ij} \quad (42)$$

$$\dot{\epsilon}_y = \frac{\partial \dot{v}}{\partial y} = \alpha_y \sum_j R_{ij} V_j (\dot{v}_j - \dot{v}_i) \ell_y / r_{ij} \quad (43)$$

$$\dot{\epsilon}_z = \frac{\partial \dot{w}}{\partial z} = \alpha_z \sum_j R_{ij} V_j (\dot{w}_j - \dot{w}_i) \ell_z / r_{ij} \quad (44)$$

$$\begin{aligned} \dot{\gamma}_{xy} &= \frac{\partial \dot{u}}{\partial y} + \frac{\partial \dot{v}}{\partial x} \\ &= \sum_j R_{ij} V_j [\alpha_y (\dot{u}_j - \dot{u}_i) \ell_y + \alpha_x (\dot{v}_j - \dot{v}_i) \ell_x] / r_{ij} \end{aligned} \quad (45)$$

$$\begin{aligned} \dot{\gamma}_{xz} &= \frac{\partial \dot{u}}{\partial z} + \frac{\partial \dot{w}}{\partial x} \\ &= \sum_j R_{ij} V_j [\alpha_z (\dot{u}_j - \dot{u}_i) \ell_z + \alpha_x (\dot{w}_j - \dot{w}_i) \ell_x] / r_{ij} \end{aligned} \quad (46)$$

$$\begin{aligned} \dot{\gamma}_{yz} &= \frac{\partial \dot{v}}{\partial z} + \frac{\partial \dot{w}}{\partial y} \\ &= \sum_j R_{ij} V_j [\alpha_z (\dot{v}_j - \dot{v}_i) \ell_z + \alpha_y (\dot{w}_j - \dot{w}_i) \ell_y] / r_{ij} \end{aligned} \quad (47)$$

$$\begin{aligned} \omega_{xy} &= \frac{1}{2} \left( \frac{\partial \dot{v}}{\partial x} - \frac{\partial \dot{u}}{\partial y} \right) \\ &= \sum_j R_{ij} V_j [\alpha_x (\dot{v}_j - \dot{v}_i) \ell_x - \alpha_y (\dot{u}_j - \dot{u}_i) \ell_y] / 2r_{ij} \end{aligned} \quad (48)$$

$$\begin{aligned} \omega_{xz} &= \frac{1}{2} \left( \frac{\partial \dot{u}}{\partial z} - \frac{\partial \dot{w}}{\partial x} \right) \\ &= \sum_j R_{ij} V_j [\alpha_z (\dot{u}_j - \dot{u}_i) \ell_z - \alpha_x (\dot{w}_j - \dot{w}_i) \ell_x] / 2r_{ij} \end{aligned} \quad (49)$$

$$\begin{aligned} \omega_{yz} &= \frac{1}{2} \left( \frac{\partial \dot{v}}{\partial y} - \frac{\partial \dot{w}}{\partial z} \right) \\ &= \sum_j R_{ij} V_j [\alpha_y (\dot{w}_j - \dot{w}_i) \ell_y - \alpha_z (\dot{v}_j - \dot{v}_i) \ell_z] / 2r_{ij} \end{aligned} \quad (50)$$

There is a change in nomenclature for the 3D algorithm when compared to the 2D algorithm. For the 2D strain rates and rotational rates in Eq. (7)–(11),  $\dot{u}_i$  and  $\dot{v}_i$  are the velocities corresponding to the  $x$  (radial) and  $z$  (axial) directions, respectively. For the 3D algorithm of Eqs. (42)–(50), the  $x$ ,  $y$ , and  $z$  velocities are represented by  $\dot{u}_i$ ,  $\dot{v}_i$ , and  $\dot{w}_i$ , respectively.

The three geometry factors are obtained in the same manner as for the 2D geometry, and are expressed as

$$\alpha_x = \frac{1}{\sum_j R_{ij} V_j \ell_x^2} \quad (51)$$

$$\begin{aligned} \alpha_y &= \frac{1}{\sum_j R_{ij} V_j \ell_y^2} \\ \alpha_z &= \frac{1}{\sum_j R_{ij} V_j \ell_z^2} \end{aligned}$$

There is an algorithm in there is only axis representation geometry factors. As discussed the geometry may also change examine the develop a so. Finally, the

$$p_x^j = -R_{ij} V_j$$

$$p_y^j = -R_{ij} V_j$$

$$p_z^j = -R_{ij} V_j$$

#### 4 Comparison

The following for the 3D stress the  $x$  direction

$$\dot{\epsilon}_x = -\beta_x \sum_j$$

$$p_x^j = W'_{ij} V_j$$

where  $W'_{ij}$  is the  $\beta$  factors

$$\beta_x = \frac{1}{\sum_j W'_{ij}}$$

$$\beta_y = \frac{1}{\sum_j W'_{ij}}$$

$$\beta_z = \frac{1}{\sum_j W'_{ij}}$$

The terms  $\beta$  Smoothing method (with assumed to be

If each of equated, the

$$\alpha_x R_{ij} / r_{ij} =$$

In a similar equated, the

$$\alpha_y = \frac{1}{\sum_j R_{ij} V_j \ell_y^2} \quad (52)$$

$$\alpha_z = \frac{1}{\sum_j R_{ij} V_j \ell_z^2} \quad (53)$$

There is an additional complexity in applying the GPA algorithm in 3D geometry. For 2D axisymmetry geometry there is only one possible orientation of axes because the  $z$  axis represents the axis of rotation. For 3D geometry the geometry factors are determined along the three principal axes. As discussed previously, if the axes are changed, then the geometry factors may change and the computed results may also change. Additional work is being performed to examine the magnitude of this potential problem and to develop a solution.

Finally, the nodal forces are as follows:

$$P_x^{ij} = -R_{ij} V_i V_j [\alpha_x (\sigma_i^x - Q_{ij}) \ell_x + \alpha_y \tau_i^{xy} \ell_y + \alpha_z \tau_i^{xz} \ell_z] / r_{ij} \quad (54)$$

$$P_y^{ij} = -R_{ij} V_i V_j [\alpha_y (\sigma_i^y - Q_{ij}) \ell_y + \alpha_x \tau_i^{xy} \ell_x + \alpha_z \tau_i^{yz} \ell_z] / r_{ij} \quad (55)$$

$$P_z^{ij} = -R_{ij} V_i V_j [\alpha_z (\sigma_i^z - Q_{ij}) \ell_z + \alpha_x \tau_i^{xz} \ell_x + \alpha_y \tau_i^{yz} \ell_y] / r_{ij} \quad (56)$$

#### 4

##### Comparison of the GPA and SPH methods

The following two equations represent SPH expressions for the 3D strain rate in the  $x$  direction and the 3D force in the  $x$  direction [10].

$$\dot{\epsilon}_x = -\beta_x \sum_j W'_{ij} V_j (\dot{u}_j - \dot{u}_i) \ell_x \quad (57)$$

$$P_x^{ij} = W'_{ij} V_i V_j [\beta_x (\sigma_i^x - Q_{ij}) \ell_x + \beta_y \tau_i^{xy} \ell_y + \beta_z \tau_i^{xz} \ell_z] \quad (58)$$

where  $W'_{ij}$  is the derivative of the smoothing function and the  $\beta$  factors are expressed as

$$\beta_x = \frac{-1}{\sum_j W'_{ij} V_j r_{ij} \ell_x^2} \quad (59)$$

$$\beta_y = \frac{-1}{\sum_j W'_{ij} V_j r_{ij} \ell_y^2} \quad (60)$$

$$\beta_z = \frac{-1}{\sum_j W'_{ij} V_j r_{ij} \ell_z^2} \quad (61)$$

The terms  $\beta_x W'_{ij}$ ,  $\beta_y W'_{ij}$ , and  $\beta_z W'_{ij}$  are the Normalized Smoothing Functions (NSF) [7, 10]. For the standard SPH method (without NSF), the  $\beta$  factors are effectively assumed to be unity.

If each of the terms in strain rate Eqs. (42) and (57) are equated, the result is that

$$\alpha_x R_{ij} / r_{ij} = -\beta_x W'_{ij} \quad (62)$$

In a similar manner, if force Eqs. (54) and (58) are equated, the results are

$$\alpha_x R_{ij} / r_{ij} = -\beta_x W'_{ij} \quad (63)$$

$$\alpha_y R_{ij} / r_{ij} = -\beta_y W'_{ij} \quad (64)$$

$$\alpha_z R_{ij} / r_{ij} = -\beta_z W'_{ij} \quad (65)$$

The three simple relationships of Eqs. (63)–(65) mean that it is possible to convert an existing SPH computational algorithm (with NSF), to a GPA computational algorithm, with the three simple substitutions in Eqs. (63)–(65).

#### 5

##### Smoothing

One of the problems associated with SPH is that instabilities can form in tensile stress regions. Swegle, et al. [18] showed that tensile instabilities can occur when  $W''_{ij} > 0$ , where  $W''_{ij}$  is the second derivative of the smoothing function,  $W_{ij}$ . This condition occurs when  $|W'_{ij}|$  decreases with increasing distance  $r_{ij}$ . Looking back at the relationships of Eqs. (63)–(65), it can be seen that the GPA method can also experience tensile instabilities, both for fixed connectivity ( $R_{ij} = 1.0$ ) and variable connectivity ( $R_{ij} = 1 - v_{ij}/2$ ).

There have been some recent efforts to reduce or eliminate the tensile instabilities by introducing various forms of conservative smoothing [17, 19, 20]. The smoothing algorithm presented in this paper smoothes the velocity fields, and it also conserves momentum and energy.

The distance-weighted average  $x$  velocity of neighbor nodes  $j$  is

$$\bar{u}_j = \frac{\sum_j R_{ij} \dot{u}_j}{\sum_j R_{ij}} \quad (66)$$

where  $R_{ij}$  is the same distance weighting function in Eq. (1) and  $\dot{u}_j$  is the  $x$  velocity of node  $j$ . The incremental velocity change imposed on node  $i$  (to smooth the velocity field) is then expressed as

$$\Delta \dot{u}_i = C_{cs} (\bar{u}_j - \dot{u}_i) \quad (67)$$

where  $C_{cs}$  is a smoothing coefficient less than unity.

To conserve momentum the velocities of neighbor nodes  $j$  must also be adjusted. Conservation of momentum requires

$$M_i \Delta \dot{u}_i + \sum_j M_j \Delta \dot{u}_j = 0 \quad (68)$$

Now it is assumed that

$$\Delta \dot{u}_j = \alpha_s R_{ij} \Delta \dot{u}_i \quad (69)$$

where  $\alpha_s$  is a constant to be determined.

Substituting Eq. (69) into Eq. (68) gives

$$M_i \Delta \dot{u}_i + \alpha_s \Delta \dot{u}_i \sum_j R_{ij} M_j = 0 \quad (70)$$

which can be solved for  $\alpha_s$ .

$$\alpha_s = \frac{-M_i}{\sum_j R_{ij} M_j} \quad (71)$$

substituting  $\alpha_s$  into Eq. (69) gives the velocity changes to neighbor nodes  $j$

$$\Delta \dot{u}_j = \alpha_s R_{ij} \Delta \dot{u}_i = \frac{-M_i R_{ij} \Delta \dot{u}_i}{\sum_j R_{ij} M_j} \quad (72)$$

This same procedure can be used to determine  $\Delta \dot{v}_i$  and  $\Delta \dot{v}_j$  (in 2D geometry) or  $\Delta \dot{v}_i$ ,  $\Delta \dot{v}_j$ ,  $\Delta \dot{w}_i$ , and  $\Delta \dot{w}_j$  (in 3D geometry).

The energy at each node is conserved by converting the change in kinetic energy to internal energy at that node. The incremental internal energy added to each node is

$$\Delta E_i = M_i [V_i^2 - (V_i + \Delta V_i)^2] / 2 \quad (73)$$

where  $V_i$  is the net velocity before smoothing and  $V_i + \Delta V_i$  is the net velocity after smoothing.

## Examples

The first examples are shown in Fig. 5. The four axisymmetric cylinder impact computations are run to a time of 0  $\mu$ s. The initial length and diameter of the cylinder are  $L_0 = 15.2$  mm and  $D_0 = 10.2$  mm, and the material is Armco Iron. The impact velocity is 305 m/s and the nodes on the rigid surface are not allowed to lift off the surface. All of the GPA computations use ghost (fictitious mirror image) nodes on the centerline  $z$  axis and the rigid surface to enforce zero displacement boundary conditions. The constitutive model and constants for the Armco Iron are as provided by Johnson and Cook [21].

In Fig. 5, the outline of the deformed cylinder and the plastic strain contours on the left side of the cylinder come from a finely gridded (1728 triangular elements) finite

element computation (which does not require a bond viscosity). The darkened GPA nodes are for plastic strain regions of 0.25–0.75 and 1.25–1.75. These regions correspond to the shaded regions of the finite element computation. In addition to a general comparison of the deformed shapes and strain contours, three normalized quantities are provided.  $L^*$  is the ratio of the final height of the GPA computation divided by the final height of the finite element computation, and  $D^*$  is a similar ratio for the diameter.  $PW^*$  is the ratio of the plastic work in the GPA computation divided by the plastic work in the finite element computation.

The four computations in Fig. 5 use variable nodal connectivity and a dimensionless influence distance of  $\alpha = 1.0$ . The computation in the upper left uses nodal viscosity ( $C_L = 0.2$  and  $C_Q = 4.0$ ) and no smoothing. These GPA results are in very good agreement with the finite element results, with  $L^* = 1.01$ ,  $D^* = 1.00$ , and  $PW^* = 0.99$ .

The computation in the upper right is identical to that in the upper left, except that smoothing is incorporated ( $C_{cs} = 0.1$ ) only when there are tensile pressures in the GPA nodes. This problem is generally under compression, but the smoothing has some effect on the results, with  $L^* = 1.03$ ,  $D^* = 0.94$  and  $PW^* = 0.95$ .

The lower left computation uses bond viscosity ( $C_L = 0.2$  and  $C_Q = 4.0$ ) instead of nodal viscosity, and it can be seen that the bond viscosity introduces artificial stiffness to the GPA computation, with  $L^* = 1.09$ ,  $D^* = 0.89$ , and  $PW^* = 0.79$ . The reduced plastic work ( $PW^* = 0.79$ ) means that about 21 percent of the plastic work is replaced by artificial viscosity work. While this cylinder impact problem will run with nodal viscosity

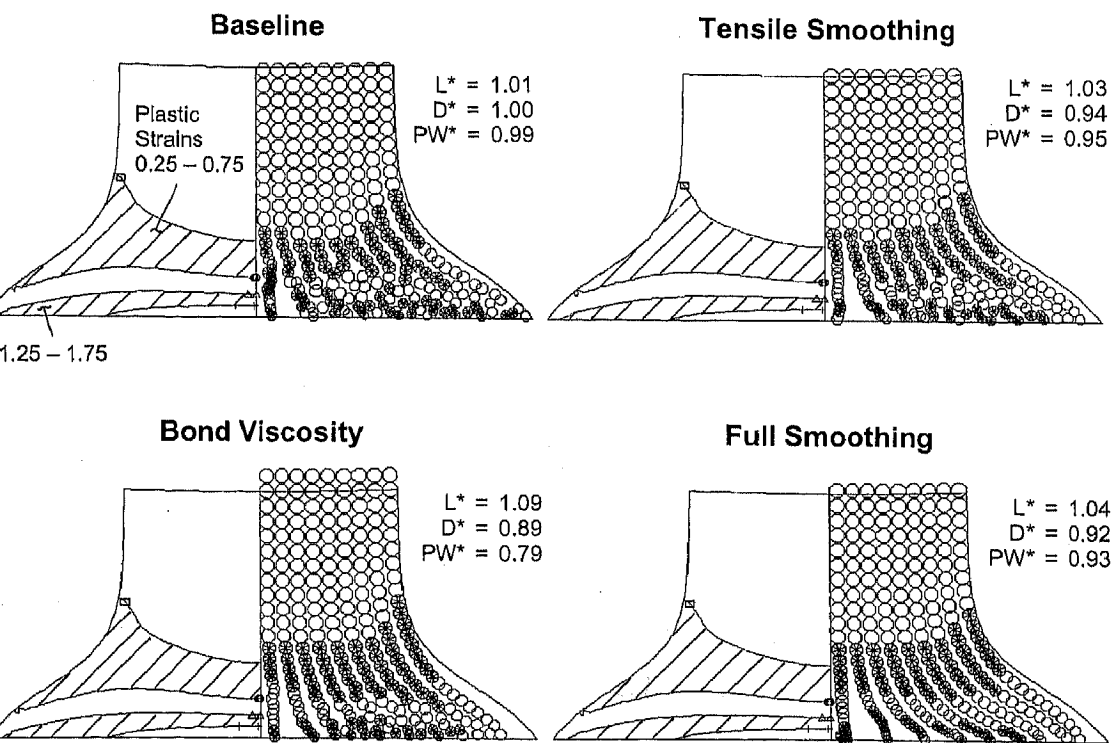
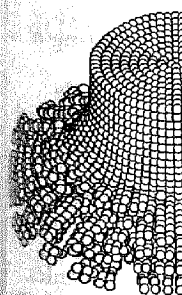
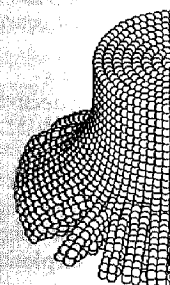


Fig. 5. 2D Cylinder impact computations

Base



Bond V



only, some other viscosity. A more viscosity effects

The lower rig smoothing ( $C_{cs}$  Here the full sm in artificial stiff smoothing), wit

The same cyl shown in Fig. 6. 2D cases in Fig. There are some edges of the imp begin as tensile distance exceed ical fracture is c passed between fracture. It can l less numerical f alized conclusio or velocity smo cosity or veloci and others may options.

Figure 7 sho of a 2024 T351 1.27 mm diam thick by 102 m models are pro problem begins

substituting  $\alpha_s$  into Eq. (69) gives the velocity changes to neighbor nodes  $j$

$$\Delta \dot{u}_j = \alpha_s R_{ij} \Delta \dot{u}_i = \frac{-M_i R_{ij} \Delta \dot{u}_i}{\sum_j R_{ij} M_j} \quad (72)$$

This same procedure can be used to determine  $\Delta \dot{v}_i$  and  $\Delta \dot{v}_j$  (in 2D geometry) or  $\Delta \dot{v}_i$ ,  $\Delta \dot{v}_j$ ,  $\Delta \dot{w}_i$ , and  $\Delta \dot{w}_j$  (in 3D geometry).

The energy at each node is conserved by converting the change in kinetic energy to internal energy at that node. The incremental internal energy added to each node is

$$\Delta E_i = M_i [V_i^2 - (V_i + \Delta V_i)^2] / 2 \quad (73)$$

where  $V_i$  is the net velocity before smoothing and  $V_i + \Delta V_i$  is the net velocity after smoothing.

## 6

### Examples

The first examples are shown in Fig. 5. The four axisymmetric cylinder impact computations are run to a time of 40  $\mu$ s. The initial length and diameter of the cylinder are  $L_0 = 15.2$  mm and  $D_0 = 10.2$  mm, and the material is Armco Iron. The impact velocity is 305 m/s and the nodes on the rigid surface are not allowed to lift off the surface. All of the GPA computations use ghost (fictitious mirror image) nodes on the centerline  $z$  axis and the rigid surface to enforce zero displacement boundary conditions. The constitutive model and constants for the Armco Iron are as provided by Johnson and Cook [21].

In Fig. 5, the outline of the deformed cylinder and the plastic strain contours on the left side of the cylinder come from a finely gridded (1728 triangular elements) finite

element computation (which does not require a bond viscosity). The darkened GPA nodes are for plastic strain regions of 0.25–0.75 and 1.25–1.75. These regions correspond to the shaded regions of the finite element computation. In addition to a general comparison of the deformed shapes and strain contours, three normalized quantities are provided.  $L^*$  is the ratio of the final height of the GPA computation divided by the final height of the finite element computation, and  $D^*$  is a similar ratio for the diameter.  $PW^*$  is the ratio of the plastic work in the GPA computation divided by the plastic work in the finite element computation.

The four computations in Fig. 5 use variable nodal connectivity and a dimensionless influence distance of  $\alpha = 1.0$ . The computation in the upper left uses nodal viscosity ( $C_L = 0.2$  and  $C_Q = 4.0$ ) and no smoothing. These GPA results are in very good agreement with the finite element results, with  $L^* = 1.01$ ,  $D^* = 1.00$ , and  $PW^* = 0.99$ .

The computation in the upper right is identical to that in the upper left, except that smoothing is incorporated ( $C_{cs} = 0.1$ ) only when there are tensile pressures in the GPA nodes. This problem is generally under compression, but the smoothing has some effect on the results, with  $L^* = 1.03$ ,  $D^* = 0.94$  and  $PW^* = 0.95$ .

The lower left computation uses bond viscosity ( $C_L = 0.2$  and  $C_Q = 4.0$ ) instead of nodal viscosity, and it can be seen that the bond viscosity introduces artificial stiffness to the GPA computation, with  $L^* = 1.09$ ,  $D^* = 0.89$ , and  $PW^* = 0.79$ . The reduced plastic work ( $PW^* = 0.79$ ) means that about 21 percent of the plastic work is replaced by artificial viscosity work. While this cylinder impact problem will run with nodal viscosity

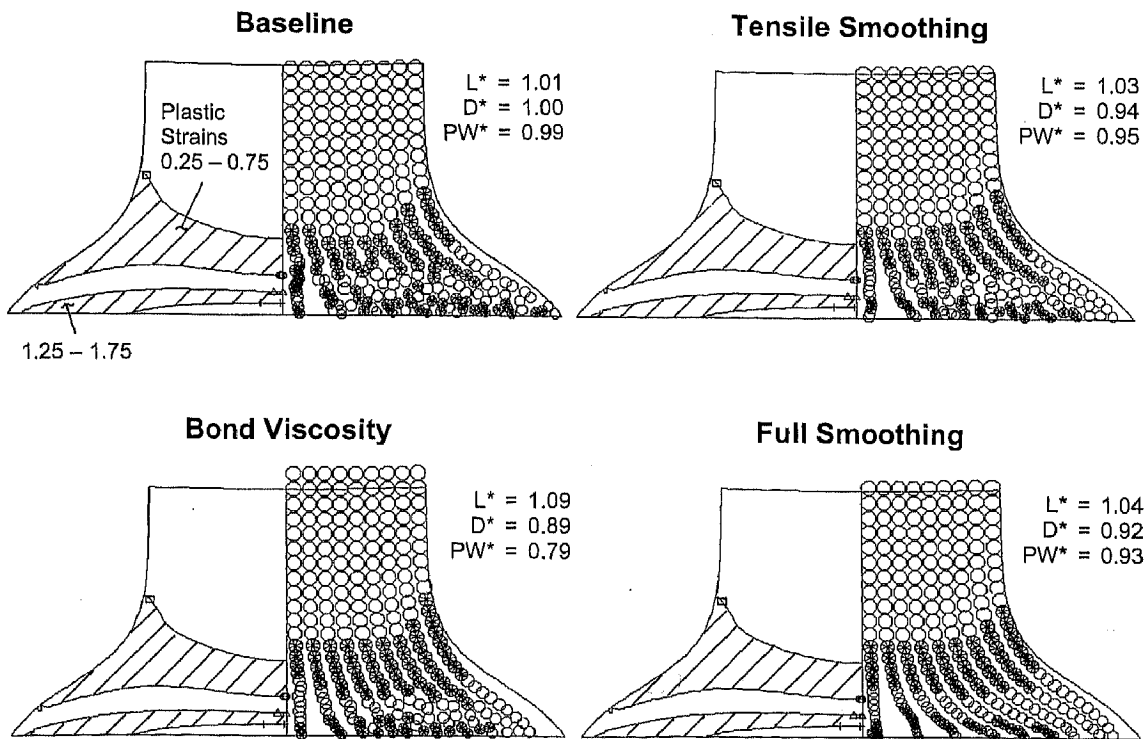


Fig. 5. 2D Cylinder impact computations

only, so  
viscosity  
viscosity  
The l  
smoother  
Here the  
in artific  
smoother  
The s  
shown in  
2D cases  
There are  
edges of  
begin as  
distance  
ical frac  
passed b  
fracture  
less num  
alized co  
or veloc  
cosity o  
and othe  
options.  
Figure  
of a 202  
1.27 mm  
thick by  
models  
problem

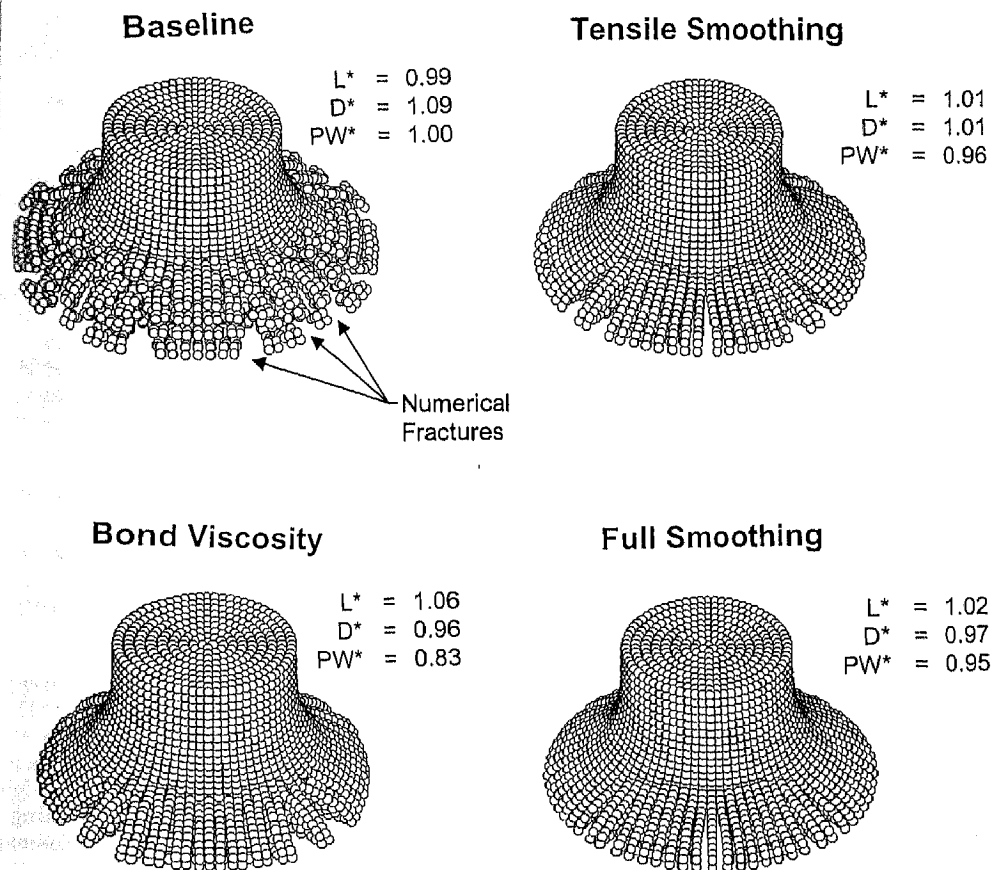


Fig. 6. 3D Cylinder impact computations

only, some other problems require some form of bond viscosity. A more comprehensive description of artificial viscosity effects is available elsewhere [10, 14].

The lower right computation uses nodal viscosity and smoothing ( $C_{cs} = 0.1$ ) for both compression and tension. Here the full smoothing introduces only a slight increase in artificial stiffness (when compared to the tensile smoothing), with  $L^* = 1.04$ ,  $D^* = 0.92$ , and  $PW^* = 0.93$ .

The same cylinder impact problem in 3D geometry is shown in Fig. 6. The four 3D cases are identical to the four 2D cases in Fig. 5, and the trends of the results are similar. There are some numerical fractures around the outer edges of the impacted end of the cylinders. These probably begin as tensile instabilities, and when the separation distance exceeds the radius of influence, then the numerical fracture is complete because no stresses and forces are passed between the nodes on either side of the numerical fracture. It can be seen that the cases with smoothing have less numerical fracture. It is difficult to make any generalized conclusions regarding the use of bond viscosity and/or velocity smoothing. Some problems require bond viscosity or velocity smoothing to simply obtain a solution and others may benefit from the use of either of these options.

Figure 7 shows a hypervelocity impact ( $V_0 = 4000$  m/s) of a 2024 T351 aluminum cylinder (0.64 mm length by 1.27 mm diameter) onto an Armco Iron plate (25.4 mm thick by 102 mm diameter). The strength and fracture models are provided by Johnson and Cook [21, 22]. This problem begins with all nodes having fixed connectivity

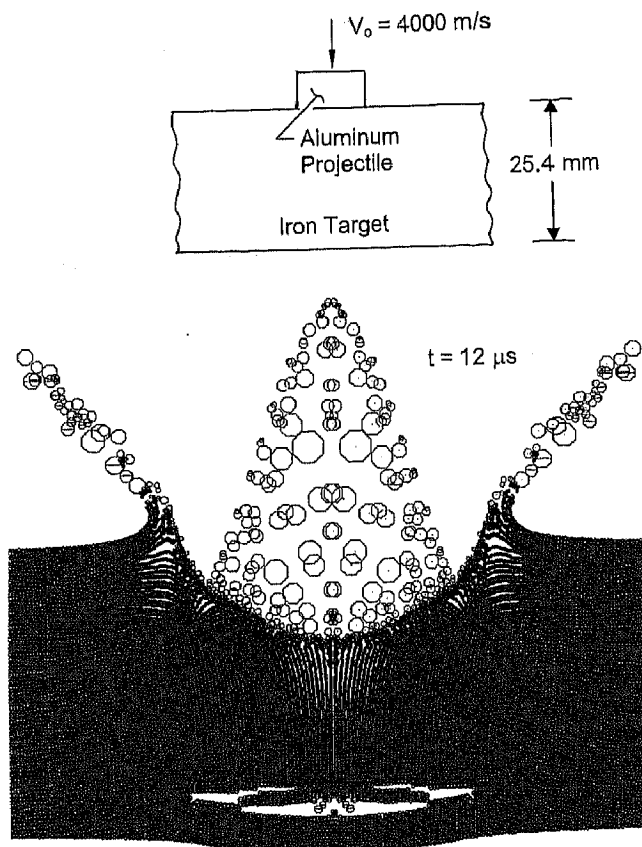


Fig. 7. Hypervelocity impact computation

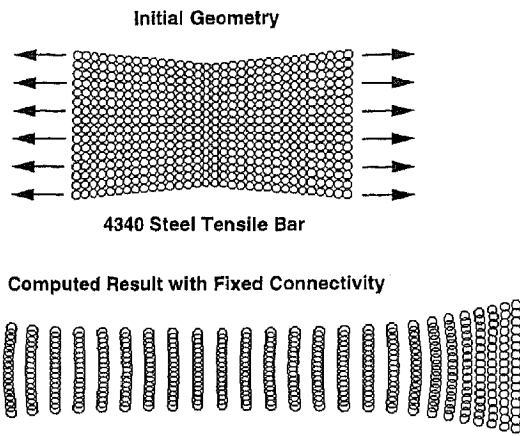


Fig. 8. Stretched tensile bar computation

(with eight neighbor nodes). Then, as the plastic strains in the nodes reach a user-input value of 0.1, the connectivity is automatically switched from fixed to variable. This result is essentially identical to that obtained only with variable connectivity, except that the CPU time was reduced by about 40 percent (because of the reduced searching time for nearest neighbors). The CPU savings is problem dependent, and it can be significant for problems with many nodes. Even though this problem does not utilize any special boundary or interface algorithms, the solution is robust and a spall region is formed as expected. It should be noted that the spall region is a true material fracture and not a numerical fracture. The fracture model fails the nodes around the spall region when the strains are small (but when the tensile pressures are high). After failure the GPA algorithm allows the fracture to open in a physically realistic manner.

Figure 8 shows a 4340 steel [21] tensile bar that is stretched to over twice its original length. It is in 2D axisymmetric geometry with fixed connectivity (eight neighbor nodes). This problem requires smoothing ( $C_{es} = 0.1$ ) to maintain a stable solution. This problem demonstrates that large tensile strains can be represented in a stable computation, but an assessment of accuracy (for this and other problems) has not yet been performed.

## 7

### Summary and conclusions

A new Generalized Particle Algorithm (GPA) has been presented for high velocity impact and other dynamics problems. It can be used with either fixed or variable nodal connectivity. The algorithm includes a smoothing option to enhance numerical stability for problems involving tensile strains and stresses. An assessment of boundary and material interface accuracy indicates some errors, but these are well understood and can be corrected with future work. The example problems demonstrate the wide range of capabilities for this technique, and the initial results are encouraging. Additional work is required to evaluate accuracy and stability for various applications.

### References

- [1] Computer Methods in Applied Mechanics and Engineering, Vol. 139, 1996
- [2] Lucy LB (1977) A numerical approach to the testing of fusion process. *The Astronomical J.* 88:1013-1024
- [3] Gingold RA, Monaghan JJ (1977) Smoothed particle hydrodynamics: Theory and application to non-spherical stars. *Monthly Notices Royal Astronomy Society* 181:375-389
- [4] Benz W (1989) Smooth particle hydrodynamics: A review. Harvard-Smithsonian Center for Astrophysics (Preprint 2884)
- [5] Monaghan JJ (1992) Smoothed particle hydrodynamics. *Annual Review Astronomy and Astrophysics* 30:543-574
- [6] Libersky LD, Petschek AG (1990) Smooth particle hydrodynamics with strength of materials. *Advances in the Free Lagrange Method, Lecture Notes in Physics* 395:248-257
- [7] Johnson GR, Beissel SR (1996) Normalized smoothing functions for SPH impact computations. *Int. J. Num. Meth. Eng.* 39:2725-2741
- [8] Johnson GR (1983) Status of EPIC codes, material characterization and New Computing Concepts at Honeywell. *Computational Aspects of Penetration Mechanics, Lecture Notes in Engineering* 3:24-35
- [9] Johnson GR, Stryk RA, Dodd JG (1986) Dynamic Lagrangian computations for solids with variable connectivity for severe distortions. *Int. J. Num. Meth. Eng.* 23:509-522
- [10] Johnson GR, Stryk RA, Beissel SR (1996) SPH for high velocity impact computations. *Comp. Meth. Appl. Mech. Eng.* 139:347-373
- [11] Swegle JW, Attaway SW (1995) On the feasibility of using smoothed particle hydrodynamics for underwater explosion calculations. *Comput. Mech.* 17:151-168
- [12] Johnson GR (1979) Dynamic response of axisymmetric solids subjected to impact and spin. *AIAA J.* 17(9):975-979
- [13] Monaghan JJ, Gingold RA (1983) Shock simulation by the particle method SPH. *J. Comput. Phys.* 52:374-389
- [14] Johnson GR (1996) Artificial viscosity effects for SPH impact computations. *Int. J. Impact Eng.* 18:477-488
- [15] VonNeumann J, Richtmeyer RD (1950) A method for the numerical calculation of hydrodynamic shocks. *J. Appl. Phys.* 21:232-237
- [16] Walsh RT (1972) Finite difference methods. In: Chou PC, Hopkins AK (eds.) *Dynamic Response of Materials to Intense Impulsive Loading*
- [17] Randles PW, Libersky LD (1996) Smoothed particle hydrodynamics: Some recent improvements and applications. *Comp. Meth. Appl. Mech. Eng.* 139:375-408
- [18] Swegle JW, Attaway SW, Heinsteins MW, Mello FJ, Hicks DL (1994) An analysis of smoothed particle hydrodynamics. SAND93-2513, Sandia National Laboratories, Albuquerque, NM
- [19] Wen Y, Hicks DL, Swegle JW (1994) Stabilizing SPH with conservative smoothing. SAND94-1932, Sandia National Laboratories, Albuquerque, NM
- [20] Guenther C, Hicks DL, Swegle JW (1994) Conservative smoothing versus artificial viscosity. SAND94-1853, Sandia National Laboratories, Albuquerque, NM
- [21] Johnson GR, Cook WH (1983) A constitutive model and data for metals subjected to large strains, high strain rates, and high temperatures. *Proceedings of Seventh International Symposium on Ballistics, The Hague, The Netherlands*, pp. 541-547
- [22] Johnson GR, Cook WH (1985) Fracture characteristics of three metals subjected to various strains, strain rates, temperatures and pressures. *Eng. Fract. Mech.* 21:31-48

Multip  
of adi  
s.

Abstract  
(RKPM),  
the analys  
grangian f  
tion are en  
band form  
multiresol  
then accor  
scale comp  
adaptive p  
suggested  
of transien  
inally pre  
meshfree  
high-scale

Introduction  
In recent  
been deve  
lems to w  
not adequ  
Reproduci  
et al. (199  
method by  
hp-Cloud  
of Unity M  
and Melen  
(PPM) by  
(NEM) by  
gral Equat  
and Mesh  
by Atluri  
feature of  
tion is con  
employing  
are called  
tages over  
remeshing  
FEM is us  
deformation

S. Jun, S. Im  
Department  
Korea Adva  
Science Tow

The postdoc  
funds (KOS



# Computational Mechanics

25

2-3

Solids, Fluids, Engineered Materials, Aging Infrastructure,  
Molecular Dynamics, Heat Transfer, Manufacturing  
Processes, Optimization, Fracture & Integrity



Special issue

Meshfree Particle Methods

Guest Editors: J.-S. Chen and K.W. Liu

Springer

Volume 25 • Number 2-3 • March 2000

Available  
online

<http://link.springer.de>  
<http://link.springer-nj.com>

	<b>Editorial</b>	99
	<b>Originals</b>	
Li S, Hao W, Liu WK:	Numerical simulations of large deformation of thin shell structures using meshfree methods	102
Wagner GJ, Liu WK:	Turbulence simulation and multiple scale subgrid models	117
Chen JS, Wang H-P, Yoon S, You Y:	Some recent improvements in meshfree methods for incompressible finite elasticity boundary value problems with contact	137
Kim NH, Choi KK, Chen JS, Park YH:	Meshless shape design sensitivity analysis and optimization for contact problem with friction	157
Atluri SN, Zhu T-L:	The meshless local Petrov-Galerkin (MLPG) approach for solving problems in elasto-statics	169
Atluri SN, Sladek J, Sladek V, Zhu T:	The local boundary integral equation (LBIE) and its meshless implementation for linear elasticity	180
Breitkopf P, Touzot G, Villon P:	Double grid diffuse collocation method	199
Bueche D, Sukumar N, Moran B:	Dispersive properties of the natural element method	207
Saigal S, Barry W:	A slices based Element Free Galerkin formulation	220
Christon MA, Roach DW:	The numerical performance of wavelets for PDEs: the multi-scale finite element	230

(Continuation on cover page 4)



0178-7675(200003)25:2-3;1-D

Online edition in LINK-  
Computer, Engineering and  
Mathematics Online Library  
<http://link.springer.de>  
Indexed in Current Contents

25 (2-3) 99-317 (2000) Printed on acid-free paper