SLIDING INTERFACES WITH CONTACT-IMPACT 
IN LARGE-SCALE LAGRANGIAN COMPUTATIONS*

J.O. HALLOQUIST, G.L. GOUDREAU and D.J. BENSON 
Lawrence Livermore National Laboratory, University of California, Livermore, CA 94550, U.S.A. 

Received 7 January 1985 

The two- and three-dimensional contact algorithms used in the finite element programs developed at the Lawrence Livermore National Laboratory are described in this paper. We are interested in both static contact and dynamic impact problems and, consequently, have pursued the development of two different algorithms. The first, based on the hydrocode technology of the sixties, is implemented in our two- and three-dimensional explicit finite element codes. The second, a symmetric penalty treatment, is used in our implicit codes and is optional in the explicit codes. The penalty methods are used to obtain solutions to almost all of our structural problems but we find that the hydrocode approach is vastly superior if pressures greatly exceed the yield strength. Examples are provided to show practical applications of both approaches.

1. Introduction

The ability to treat structural problems where adjacent components may independently slide, separate, and impact along material interfaces is crucial in many varied technical fields. In weapons design, relevant problems include the structural response of gun-fired projectiles, hypervelocity impacts, and a variety of shape charge designs of either the jet or fragment type. The nuclear industry analyzes the impact of shipping casks containing radioactive materials, pipe-to-pipe impact, and soil-structure interaction problems. Crashworthiness in the automotive and aircraft industry is another important area, as is biomechanics. The contact-impact capability has always been an important aspect of both the implicit and explicit finite element codes [1–4] developed at the Lawrence Livermore National Laboratory, where nearly all our applications depend on it. Our codes use well over 1000 CPU hours of LLNL Cray time per year, but considerably more would be used if it were available to the structural analysts. Our need for more analyses leads us to continually improve the speed and efficiency of the solution algorithms, relying to a large extent on ideas from the literature and our own research, including a considerable amount of numerical experimentation. Today, we can solve most two- and three-dimensional contact-impact, finite strain, plasticity problems of interest to us with almost complete confidence that no insurmountable numerical difficulties will arise.

The development of reliable implicit codes to do contact problems requires a sound global strategy. We have implemented modified Newton, full Newton, and quasi-Newton iteration schemes; the latter of which includes the BFGS [5] and Broyden’s [6] methods. In our experience, the quasi-Newton line searches also contribute to the fast convergence of the

*This work was performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under contract number W-7405-Eng-48.

modified and full Newton schemes. In three dimensions, we have ruled out full Newton as a viable option due to the extreme expense of reforming and factoring a large stiffness matrix for each iteration. We now regularly use quasi-Newton methods with frequent reformations of the stiffness matrix. By optimally coding and fully vectorizing the stress divergence, the expense of equilibrium iterations including line search has been minimized.

Explicit hydrocodes use the viscosity method [7] to resolve the details of shock wave propagation through the mesh with small time-step sizes dictated by the Courant stability limit. In three dimensions, our typical applications use 20,000 elements, 10,000 to 100,000 time steps, with 20 to 30 percent of the nodal points contained in the contact surface definitions. Efficient, cheap algorithms are needed to make such problems affordable. We use primitive elements (one stress point hexahedrons), control the zero energy modes by the simplest possible hourglass control [8], and have an efficient and reliable contact-impact algorithm. With faster and larger computers than presently available, we would quickly expand our three-dimensional analyses to a point where the resolution is roughly equivalent to our two-dimensional counterparts, i.e., by a factor of 10 to 50; therefore the need for efficient algorithms will continue for some time.

In developing general-purpose contact algorithms for three-dimensional solutions, we quickly realized that the rigorous extension of the two-dimensional hydrocode algorithms, though feasible, would be too costly. Our earlier experience has shown that Lagrange multiplier methods [9, 10] did not necessarily preserve a smooth force distribution across interfaces. The lack of a smooth force field excited the zero energy modes in the primitive elements in nearly all the contact-impact problems we solved. Furthermore, these modes usually stopped explosive-metal interaction computations, a major application for two-dimensional hydrocodes, early in the solution. Thus, we were left with the penalty method as the method of choice. We use the penalty approach in our two- and three-dimensional implicit and explicit codes, but, because it is not always suitable for applications involving high explosives, we retain an approach in our explicit codes that is based in part on the hydrocode methodology developed several decades ago.

In the development which follows, we shall first discuss the hydrocode sideline methodology developed in the 60's. Our DYNA2D algorithm, based on this work, has been extended to include impact and separation along the sideline. We also implemented this methodology in three dimensions but without the DYNA2D extensions. In Section 3, we discuss the penalty method. In the last section, we present a few applications.

2. Hydrocode methodology

In two dimensions, the contact surfaces along a sideline appear as lines in the y-z plane where y is the horizontal axis, and z is the vertical axis and the axis of revolution in axisymmetric problems. These lines will be referred to as the master and slave lines, respectively. Nodal points along the slave and master lines will be referred to as slave and master nodes, respectively. Likewise, line segments joining adjacent slave and master nodes will be called slave and master segments. Elements that have at least one side that is a slave segment are called slave elements, and master elements are similarly defined. Fig. 1 shows a typical interface. Slave nodes are constrained to slide on or close to the master line unless...
Figure 1. Typical interface.

In the early sliding line algorithms discussed below, these lines were not allowed to separate. Designation of master and slave lines is required as input for the sliding line definitions in most hydrocodes. A truly symmetric treatment such as used in our penalty algorithm would make this distinction irrelevant, but we are aware of no published treatments that are symmetric in the hydrocode literature.

In the finite element computer implementation of the hydrocode algorithms, no changes to the explicit solution scheme are required since it is possible to organize the algorithm such that the logic external to the interface treatment applies to each node including the slave and master nodes. The additional subroutines are called once every time step following the calculation of the accelerations but prior to updating the velocity vector.

The definition of the slave and master lines is accomplished in DYNA2D by the user who provides a list of nodal point numbers of the nodes lying on these lines. These numbers are given in the order they appear as one moves along the interface keeping the slave line on the left. Slave elements are identified in the initialization phase of DYNA2D by a search of the element connectivities and are ordered such that the first elements correspond to the first slave node, the second element to the second slave node, and so on.

In the sections which follow we will describe the algorithms of HEMP [11], TOODY [12], TENSOR [13], as well as our own more recent efforts in DYNA2D which are based on this earlier work.

21. HEMP finite difference code

In 1964, Wilkins published the HEMP finite difference equations based on the integral difference method [14]. A logically regular mesh was used made from a grid of intersecting \( j \) and \( k \) lines with sliding lines permitted along \( k \) lines. A weakly coupled sliding line algorithm was given (the mass in the slave elements was ignored), but modifications were clearly outlined for incorporating strong coupling. The actual modifications were included in a later revision of the original document [15]. In Wilkins' approach, a pressure boundary condition is applied to each segment of the master surface based on the stress state in the slave elements across from the center of the master element containing the segment. The pressure, \( p_i \) for the \( l \)th master segment is given by (see Fig. 2)

\[
p_l = (1 - \alpha) p_k + \alpha p_{k+1},
\]
where $p_k$ and $p_{k+1}$ are the normal stress components in slave elements $k$ and $k+1$;

$$p_k = -(\sigma_{yy}^k \sin^2 \theta_{Ll+1} + \sigma_{zz}^k \cos^2 \theta_{Ll+1} - \sigma_{yz}^k \sin 2\theta_{Ll+1}),$$

$$p_{k+1} = -(\sigma_{yy}^{k+1} \sin^2 \theta_{Ll+1} + \sigma_{zz}^{k+1} \cos^2 \theta_{Ll+1} - \sigma_{yz}^{k+1} \sin 2\theta_{Ll+1}),$$

and $\theta_{Ll+1}$ is the included angle taken counterclockwise between the $y$-axis and the line from $l$ to $l+1$. The parameter $\alpha$ is a dimensionless distance,

$$\alpha = L'/L,$$

where $L$ is the distance between the element centers of slave elements $k$ and $k+1$ and $L'$ is the distance from the center of slave element $k$ to the intersection of a line drawn from the center of the master element $l$ normal to segment $l$ with the line drawn between the slave element centers. After repeating this procedure for each master segment, the entire surface is accelerated as a free surface with a pressure boundary condition.

To attain the goal of a strongly coupled algorithm it is necessary to include the mass of the slave surface. A constant factor, $z$, is defined for the entire slideline and is used to scale the normal component of the acceleration,

$$z = \left(\sum_{i=1}^{nm} M_i^n + \sum_{i=1}^{ns} M_i^s\right) / \sum_{i=1}^{nm} M_i^n.$$  

In (2.4), $nm$ and $ns$ are the total number of master and slave elements along the slideline, and $M_i^n$ and $M_i^s$ are the total mass in the $i$th master and slave elements, respectively. The master node accelerations are rotated into the local coordinates system of the node and the normal acceleration is scaled to account for the added mass of the slave surface.

In the local coordinate system from $l-1$ to $l$, the tangential, pressure distribution, and acceleration vectors are

$$\dot{y}_l^n = \begin{bmatrix} \dot{x}_l^n \\ \dot{y}_l^n \end{bmatrix},$$

$$\ddot{y}_l^n = \begin{bmatrix} \ddot{x}_l^n \\ \ddot{y}_l^n \end{bmatrix}.$$ 

The final step is to scale the vectors tangentially, tangentially, and normal to the slideline.

The slave node accelerations can be scaled by the $n+1$ configuration. Velocities for the $n+1$ configuration can be calculated with time-step size

$$\dot{y}_{l+1/2}^n = \dot{y}_l^n + \Delta t \ddot{y}_l^n.$$ 

In early versions, the dependent factor $z$ was not valid and this required a computation of interface mass for the immediate vicinity of the contiguous slave. The boundary normal to the master slideline is in contact Jacobian. When the stress on $p_i$ cannot be determined, the $s$-TENSOR discussion section.

### 2.2. TOODY function

In the TOODY finite element code, the integral difference of $p_i$ between new mesh is determined by using a contiguous mass balance algorithm in DYNA

First we must define the mass of the contiguous elements. The mass of the master node is

$$\frac{1}{2} \frac{\sum_{i=1}^{nm} M_i^n + \sum_{i=1}^{ns} M_i^s}{\sum_{i=1}^{nm} M_i^n}.$$ 

In (2.4), $nm$ and $ns$ are the total number of master and slave elements along the slideline, and $M_i^n$ and $M_i^s$ are the total mass in the $i$th master and slave elements, respectively. The master node accelerations are rotated into the local coordinates system of the node and the normal acceleration is scaled to account for the added mass of the slave surface.
\[ a_{nl}^{n+1} = \frac{1}{z} \left( \bar{z}_l^+ \cos \theta_{l-1,l+1} - \bar{y}_l^+ \sin \theta_{l-1,l+1} \right), \]  
(2.5a)

\[ a_{nl}^{n+1} = \bar{y}_l^+ \cos \theta_{l-1,l+1} + \bar{z}_l^+ \sin \theta_{l-1,l+1}. \]  
(2.5b)

In the local coordinate system, the normal direction at point \( l \) is perpendicular to a line drawn from \( l-1 \) to \( l+1 \). The superscript, +, signifies that acceleration has been updated for the pressure distribution. The global accelerations are calculated by rotating the scaled acceleration vector back to the global frame,

\[ \bar{y}_l^{n+1} = a_{nl}^{n+1} \cos \theta_{l-1,l+1} - a_{nl}^{n+1} \sin \theta_{l-1,l+1}, \]  
(2.6a)

\[ \bar{z}_l^{n+1} = a_{nl}^{n+1} \cos \theta_{l-1,l+1} + a_{nl}^{n+1} \sin \theta_{l-1,l+1}. \]  
(2.6b)

The final step is the update of the motion of the slave surface. Each slave node \( k \) is moved tangentially, treating the master surface as a symmetry plane in the configuration at time \( n \). The slave nodes are projected back onto the master surface, after the surface is updated into the \( n+1 \) configuration, along the \( j \) line on the slave side that contains the \( k \)th slave node. Velocities for the slave nodes are then found by dividing the change in coordinates by the time-step size,

\[ y_l^{n+1/2} = (y_l^n - y_l^n)/\Delta t^{n+1/2}, \quad z_l^{n+1/2} = (z_l^{n+1} - z_l^n)/\Delta t^{n+1/2}. \]  
(2.7)

In early versions of HEMP, the constant \( z \)-factor of (2.4) was used rather than a spatially \((j)\) dependent factor. Of course, the assumption of a constant mass distribution is not generally valid and this was acknowledged by Wilkins. Another approximation involved the computation of interface pressure from (2.1) and (2.2) where only two slave elements in the immediate vicinity of the mass segment are considered whereas a weighted average of all contiguous slave elements would give a more realistic representation of \( p_l \). Also, the pressure normal to the master segment defined by rotating the planar stress is valid only in plane strain constant Jacobian elements. Ignoring the geometry of the element and the effects of hoop stress on \( p_l \) can lead to values that are too high or low. The methodology of TOODY and TENSOR discussed below overcomes these weaknesses.

2.2 TOODY finite difference

In the TOODY implementation of sliding interfaces, Bertholf and Benzley applied the integral difference method to each master node along the sliding interface. During each cycle a new mesh is defined overlaying the slave elements and matching the mesh density of the contiguous master surface. We shall briefly outline our implementation of the TOODY algorithm in DYNA2D.

First we must locate the coordinates of element \( k^* \), where the asterisk superscript denotes any element and related variables that are assumed to lie in the slave elements that overlap the master nodes. The position of point \( j^* \) in Fig. 3, for example, is determined such that the ratio of the distances is constant when \( d_{k,l} \) is the distance from point \( k \) to \( l \),
\[
\frac{d_{k,l}}{d_{k,k+1}} = \frac{d_{j,j+1}}{d_{j,j+1}}.
\]

The stresses in the newly defined slave elements are based on a length-weighted interpolation along the master surface. Consider Fig. 3 where slave element \( k \) is the first element to overlap master segment \( l \) and overlies the phony element \( k^* \). Letting \( S_l \) denote the length of the master segment \( l \), \( n_i \) denote the unit vector tangent to segment \( l \), and \( r_{l,k} \) denote the vector drawn from \( l \) to slave node \( k \), we can define \( S_{l,k} \) for each slave node:

\[
S_{l,k} = \begin{cases} 
    r_{l,k} \cdot n_i & \text{if } r_{l,k} \cdot n_i > 0, \\
    0 & \text{if } r_{l,k} \cdot n_i \leq 0,
\end{cases}
\]

\[
S_{l,k+1} = \begin{cases} 
    r_{l,k+1} \cdot n_i & \text{if } r_{l,k+1} \cdot n_i < S_l, \\
    S_l & \text{if } r_{l,k+1} \cdot n_i \geq S_l.
\end{cases}
\]

The stress contribution to slave element \( k^* \) made by the stress in slave element \( k \), \( \sigma_{ij}^\ast_{k^*} \), is designated \( \sigma_{ij}^\ast_{k,k} \) and defined by

\[
\sigma_{ij}^\ast_{k,k} = (S_{l,k+1} - S_{l,k})\sigma_{ij}^\ast/S_l.
\]

If inequality (2.9d) is satisfied, \( \sigma_{ij}^\ast_{k,k} \) determines the stress in phony element \( k^* \); otherwise, the contribution to the stress tensor of the next overlapped slave element is added. For example, consider the contribution of zone \( k+1 \) to the phony \( k^* \) zone,

\[
S_{l,k+2} = \begin{cases} 
    r_{l,k+2} \cdot n_i & \text{if } r_{l,k+2} \cdot n_i < S_l, \\
    S_l & \text{if } r_{l,k+2} \cdot n_i \geq S_l,
\end{cases}
\]

\[
\sigma_{ij}^\ast_{k,k+1} = (S_{l,k+2} - S_{l,k+1})\sigma_{ij}^\ast_{k,k+1}/S_l.
\]

If (2.11b) is unsatisfied, \( k \) again is incremented by 1, and the procedure given by (2.11)–(2.13) continues until (2.11b) is satisfied. The stress tensor for \( k^* \) becomes

\[
\sigma_{ij}^\ast = \sigma_{ij}^\ast_{k,k} + \sigma_{ij}^\ast_{k,k+1} + \cdots
\]

where \( n \) is the number of overlapped slave elements in the bulk viscous fluid, and \( \sigma_{ij}^\ast \) are the stress forces by the normal to the surface, if they are free from contact, or the relative tangential velocities.

In the update, the stress is normally on to the relative tangential velocity.

\[
\Delta x_{i}^{n+1} = \frac{\Delta t}{\rho^*}
\]

Using the configuration.

2.3. TENSOR

Of the hydrodynamic algorithm comes from the fact that over a phony slave node, the phony slave node's stress is interpolated from the stress in Fig. 3. From the relative tangential velocity relative to the slave node, the stress is interpolated from the stress in Fig. 3.
\[ \sigma_{j}^{k^*} = \sum_{i=0}^{n-1} \sigma_{j,i,k+1}^{k^*}, \]  

(2.13)

where \( n \) is the number of overlapping elements. After repeating this procedure to determine the bulk viscosity \( q^{k^*} \) and density \( \rho^{k^*} \), we proceed to the next phony element, \( k + 1^* \), and repeat the procedure.

We enter the slideline logic in DYNA2D after computing the nodal forces, and therefore the free surface accelerations of the slave and master nodes are available by dividing the nodal forces by the nodal masses. The tangential accelerations of the master nodes are calculated as if they are free surface nodes, by using (2.5b). For the normal acceleration, the master node is temporarily treated as an interior node surrounded by the master elements and the phony slave elements. A new acceleration \( \left( \tilde{\mathbf{f}}^+_i, \tilde{\mathbf{z}}^+_i \right) \) is computed, and substituted into (2.5a) with \( z = 1 \). The modified global acceleration is calculated from \( a_{i}^{n+1} \) and \( a_{n}^{n+1} \) by using (2.6).

In the update of the slave surface motion, Bertholf and Benzley projected the slave nodes normally on to the updated master surface and moved them tangentially according to their tangential velocities and accelerations (this is illustrated in Fig. 4).

\[ \Delta x_i^{n+1} = (v_i^{n+1/2} + a_i^n \Delta t^n) \Delta t^{n+1/2}. \]

(2.14)

Using the configurations at \( n \) and \( n + 1 \) the slave node velocities are then computed from (2.7).

1.3 TENSOR finite difference code

Of the hydrocode treatments discussed in this paper, the TENSOR contact-impact algorithm comes closest to being symmetric. Like the approach in TOODY, phony elements are defined that overlay the slave elements and have interpolated values of stresses and densities. Phony slave nodes are created at each master node and are assigned velocities that are interpolated from the actual slave nodes. Consider master node \( l \) and phony node \( l^* \) as shown in Fig. 3. From rigid body dynamics we can treat \( l^* \) as a particle that has a zero normal velocity relative to a rotating coordinate system with its origin at node \( l \) and write the

Fig. 4. Update of the position of slave node \( k \) from time \( n \) to \( n + 1 \) given by Bertholf and Benzley [12].
constraint equation for the normal acceleration,

\[
a_{n,r} = a_{n,i} + a_c, \quad a_c = 2\omega_l[(\dot{y}_{l+1} - \dot{y}_l) \cos \theta_{l-1,l+1} + (\dot{z}_{l+1} - \dot{z}_l) \sin \theta_{l-1,l+1}].
\]  

(2.15)

The normal direction at node \( l \) is perpendicular to a line drawn from nodes \( l - 1 \) to \( l + 1 \), \( a_c \) is the Coriolis acceleration term, and \( \omega_l \) is the angular velocity at node \( l \),

\[
\omega_l = \frac{(\dot{z}_{l+1} - \dot{z}_{l-1}) \cos \theta_{l-1,l+1} - (\dot{y}_{l+1} - \dot{y}_{l-1}) \sin \theta_{l-1,l+1}}{d_{l-1,l+1}}.
\]  

(2.16)

The expressions for the normal accelerations in terms of an interface pressure \( p \) with (2.15) and (2.16) are solved for \( p \),

\[
a_{n,i} = \frac{(f_{n,i} - \frac{1}{2}pc_i)}{m_i}, \quad \text{(2.17a)}
\]

\[
a_{n,r} = \frac{(f_{n,r} + \frac{1}{2}pc_r)}{m_r}, \quad \text{(2.17b)}
\]

\[
c_l = (z_{l+1} - z_{l-1}) \sin \theta_{l-1,l+1} + (y_{l+1} - y_{l-1}) \cos \theta_{l-1,l+1}.
\]  

(2.18)

Here, \( M_l \) and \( M_r \) are the lumped masses, and \( f_{n,i} \) and \( f_{n,r} \) are the normal components of the nodal forces at nodes \( l \) and \( l^* \) due to the internal stress states.

Equation (2.17a) permits the normal acceleration of the master surface to be updated. This procedure is repeated for each master node. Since the tangential component of the acceleration is unchanged, the global components can be found from (2.6).

Unlike the HEMP and TOODY schemes, where the slave node is moved with the master surface, the TENSOR scheme repeats the foregoing process creating phony nodes and elements on the master side to accelerate the slave surface. After the geometry is updated, slave nodes that do not lie on the master surface are projected normally to the surface. Normal components of velocity are then reset by interpolating from the master side.

2.4. DYNA2D finite element code

The implementation in DYNA2D will be discussed in detail. Ideas from HEMP, TOODY, and TENSOR underlie the current DYNA2D algorithm. Although we feel that the TENSOR scheme may be superior on some problems, it is considerably more expensive and is not easily extended to handle impact problems.

Our current interface treatment may be outlined as follows:

Step 1. Update the location of each slave node by identifying its closest master node if it is not in contact with the master surface or by identifying the first master node of the master segment on which it lies.

Step 2. For each master segment, identify the first overlapping slave element.

Step 3. Check each slave node that lies in contact with the master surface for the existence of a tensile interface force, and if the force is tensile, release it.

Step 4. Distribute one-half the mass of each slave element to the appropriate master nodes.

if both slave nodes of these slave segments distribute over more than one master segment of the master.

Step 5. When determining the value of the master.

Step 6. For each master segment, the master.

Step 7. If the velocity and acceleration from the master.

Step 8. In the next time step.

The above procedure is assumed to be repeated for each slave node. It is made that the slave nodes slide off the master surface in any case or, if the master is more than 2D.

2.4.1. Slave node

Consider the master.

Fig. 5 depicts the master.

\[
\mathbf{r}_{l-1,l+1} \quad \text{or on the slave nodes}.
\]

where \( \mathbf{r}_{l-1,l+1} \) and \( \mathbf{r}_{l-1,l+1} \) determine the

Fig. 5. Given

\[
\mathbf{r}_{l-1,l+1}
\]
If both slave nodes of the slave segment associated with the element are in contact. If neither of these slave nodes are in contact, distribute no mass. If one slave node is in contact, distribute one-half the mass lying between the contacting node and the center of the slave segment of the slave element.

**Step 5.** Whenever a slave node reaches the master surface, apply the interface conditions to determine the post-impact velocity.

**Step 6.** From the states of stress in the slave elements, compute a distributed pressure for each master segment and the equivalent nodal forces. A slave element must be in contact with the master surface to contribute force.

**Step 7.** For each slave node in contact with a master segment, compute its tangential velocity and acceleration. Interpolate the normal components of velocity and acceleration from the adjacent master nodes.

**Step 8.** For slave nodes not in contact with the master surface check for penetration during the next time step and reduce the time-step size if necessary so that no slave node penetrates.

The above steps are explained in Sections 2.4.1–2.4.9 in more detail. A typical interface is assumed to consist of \( n_s \) slave nodes, \( n_s - 1 \) slave elements, and \( n_m \) master nodes. An assumption is made that the master line is sufficiently long so that slave nodes in contact with the line will not slide off the end of the line. In practice, master line extensions are used to ensure that this is the case or, if the user prefers, the slave nodes that slide off can be treated as free surface nodes. If more than one interface exists, the procedures outlined here are repeated from each interface.

### 2.4.1. Slave node search

Consider slave node \( k \) \( (1 < k < n_s) \) and assume that a search of the master line has located the master node \( l \) \( (1 < l < n_m) \) lying closest to \( k \). If \( k \) is free, the next slave node is considered. If \( k \) is sliding on the master surface, the master segment that contains \( k \) must be determined.

Fig. 5 depicts a portion of the master surface where slave node \( k \) and master node \( l \) are labeled. Node \( k \) lies on the segment bounded by \( l - 1 \) and \( l \), called master segment \( l - 1 \), if

\[
\mathbf{r}_{l-1,l+1} \cdot \mathbf{r}_{l,k} = 0 ,
\]

or on the segment bounded by \( l \) and \( l + 1 \), called master segment \( l \), if

\[
\mathbf{r}_{l-1,l+1} \cdot \mathbf{r}_{l,k} > 0 ,
\]

where \( \mathbf{r}_{l-1,l+1} \) is the vector from \( l - 1 \) to \( l + 1 \), and, likewise, \( \mathbf{r}_{l,k} \) is the vector from \( l \) to \( k \).
2.4.2. Slave element search

This search identifies the first slave element to overlap each master segment. Pressure contributions are made by all slave elements that overlap the segment beginning with the element identified here and ending with the last element that overlaps the segment. In Fig. 6, a portion of the master line is shown in contact with the slave line.

Consider master segment \( l \). Recall that \( n_l \) is the unit vector tangent to master segment \( l \) and let \( k \ (1 \leq k < n_s) \) be the closest slave node to master node \( l \). Slave element \( k \) is identified as the first slave element to overlap master segment \( l \) if any of the following tests are satisfied:

\[
\begin{align*}
r_{l,k} \cdot n_l &\leq 0 \quad \text{for } 1 \leq k < n_s - 1, \\
r_{l,k} \cdot n_l &\leq 0 \\
r_{l,k+1} \cdot n_l &> 0 \\
r_{l,k} \cdot n_l &> 0 \quad \text{for } k = n_s - 1, \\
r_{l+1,k} \cdot n_l &< 0 \quad \text{for } k = 1.
\end{align*}
\]

(2.21)

(2.22)

If not, slave element \( k - 1 \) is identified as the first overlapping element if

\[
\begin{align*}
r_{l,k-1} \cdot n_l &< 0 \quad \text{for } 1 < k < n_s - 1, \\
r_{l,k} \cdot n_l &> 0 \\
r_{l,k-1} \cdot n_l &< 0 \quad \text{for } k = n_s - 1.
\end{align*}
\]

(2.22h)

If the tests given by (2.21) and (2.22) fail, then no slave element overlaps the master segment being considered.

2.4.3. Tensile interface force

Consider slave node \( k \) and assume that it lies in contact with master segment \( l \). If the normal acceleration of the slave node relative to the master segment is positive, a tensile force is indicated, and the slave node is set free.

The normal acceleration of node \( k \) relative to segment \( l \) is \( \Delta \ddot{a}_n \), which is defined by

\[
\Delta \ddot{a}_n = \sum_{i=1}^{n_s} a_{n_i} - \sum_{i=1}^{n_s} a_{n_i}^l,
\]

where \( a_{n_i} \) and \( a_{n_i}^l \) are the normal accelerations of slave node \( i \) and master node \( l \), respectively.

If inequalities (2.21) and (2.22) are not satisfied, then

\[
\Delta \ddot{a}_n = \sum_{i=1}^{n_s} a_{n_i} - \sum_{i=1}^{n_s} a_{n_i}^l,
\]

where \( s_{k,i} \) is the time of contact between the slave element \( k \) and the master segment \( l \).

Fig. 6. A typical portion of the master line which shows slave element \( k \) as the first element to overlap master segment \( l \).
\[ \Delta \tilde{a}_{nl} = \tilde{a}_{nl} - (1 - \alpha_k) \tilde{a}_{nl} - \alpha_k \tilde{a}_{nl+1}, \]  

(2.23)

The local coordinate \( \alpha_k \) is defined by

\[ \alpha_k = n_l \cdot r_{nk} / (n_l \cdot r_{k,l+1}). \]  

(2.24)

If \( \alpha_k < \varepsilon \) or if \( \alpha_k > 1 - \varepsilon \), where \( \varepsilon \) is a tolerance typically in the range of 0.005, the angle \( \theta_{l,t-1} \) or \( \theta_{l,t+2} \) is used in place of \( \theta_{l,t+1} \) to determine the normal in (2.23). The right superscript, \( - \), on nodal quantities denotes the values of the quantities before interface coupling is taken into account.

### 2.4.4. Addition of slave mass to master mass

The mass of the slave elements along the master surface is attributed to the appropriate master nodes. Each slave element is considered separately.

Consider the \( k \)th slave element. The mass to be distributed is given by

\[ m_k = \frac{1}{2} \rho k A_k, \]  

(2.25)

for plane strain. For axisymmetric problems, the DYNA2D Petrov–Galerkin scheme is used and, in the above equations, \( \rho k \) and \( A_k \) are the density and area of the \( k \)th slave element in the current configuration.

As indicated previously, the \( k \)th slave element corresponds to the \( k \)th slave node. The \( k \)th slave node is being tracked by the master node \( l \) and is, therefore, known to lie on master segment \( l \) bounded by master nodes \( l \) and \( l + 1 \). Let

\[ L_{k,i} = \begin{cases} s_{k,i} \cdot n_k & \text{if } s_{k,i} \cdot n_k > 0, \\ 0 & \text{if } s_{k,i} \cdot n_k \leq 0, \end{cases} \]  

(2.26)

where \( s_{k,i} \) is the vector from the \( k \)th slave node to the center of master segment \( l \), and \( n_k \) is the unit vector tangent to the slave segment \( k \) bounded by slave nodes \( k \) and \( k + 1 \),

\[ n_k = r_{k,k+1} / |r_{k,k+1}|. \]  

(2.27)

Let \( L_k \) denote the length of slave segment \( k \), i.e., \( L_k = n_k \cdot r_{k,k+1}. \) The mass attributed to master node \( l \) from slave element \( k \), \( m_{k,l} \), is given by

\[ m_{k,l} = \begin{cases} (L_k / L_k) m_k & \text{if } L_k > L_{k,i}, \\ m_k & \text{if } L_k \leq L_{k,i}. \end{cases} \]  

(2.28a)

(2.28b)

If inequality (2.28b) is satisfied, the next slave element is considered; otherwise, let

\[ L_{k,i+1} = \begin{cases} s_{k,i+1} \cdot n_k & \text{if } s_{k,i+1} \cdot n_k < L_k, \\ L_k & \text{if } s_{k,i+1} \cdot n_k \geq L_k, \end{cases} \]  

(2.29a)

(2.29b)

where \( s_{k,i+1} \) is the vector from the \( k \)th slave node to the center of master segment \( l + 1 \). The
mass attributed to master node \( l + 1 \) from slave element \( k \) is given by

\[
m_{k,l+1} = \frac{L_{k,l+1} - L_{k,l}}{L_k} m_k.
\] (2.30)

If (2.29b) is satisfied, the next slave element is considered; otherwise, the above procedure, using (2.29) and (2.30) with subscript \( l \) increased by one is repeated until (2.29b) is satisfied. This procedure is illustrated in Fig. 7.

The foregoing procedure is modified slightly whenever a void exists between the master surface and the \( k \)th element. If neither slave node lies on the master surface, the next slave element is immediately considered. If only one slave node is in contact, \( L_k \) is set to one-half of the length of slave segment, and, if node \( k \) is in contact, no other changes are required. If node \( k + 1 \) is in contact, vector \( s_{k,l+1} (i = 1, 2, \ldots) \) is computed with its origin at the center of the slave segment.

2.4.5. Momentum calculations

Assume slave node \( k \) impacts the master line. By conserving momentum, the post-impact velocity of the master line can be obtained. The amount of mass associated with slave node \( k \) is taken as one-half of the mass included from the center of slave segment \( k - 1 \) of slave element \( k - 1 \) to the center of slave segment \( k \) of slave element \( k \). Let \( m_{k,l} \) denote the mass associated with slave node \( k \) that is attributed to master node \( l \). The mass that is distributed as described above is given by

\[
v_{l,i}^+ = \frac{m_{k,l}}{M_i} \quad i = 1, 2, \ldots
\]

\[
M_i
\]

is the mass of node \( i \). Defining the vector

\[
v_{l,i}^- = \frac{m_{k,l}}{M_i} \quad i = 1, 2, \ldots
\]

the global post-impact velocity vector

\[
y_{l}^- = \frac{m_{k,l}}{M_i} \quad i = 1, 2, \ldots
\]

2.4.6. Master Surface

The pressure on the master surface in contact with the slave nodes results in normal stresses for the master surface as well. However, since we need only contact force along the master surface.

With reference to Fig. 7, the impact of slave element \( k \) is completely transferred to the master element \( k \) perpendicularly

\[
\rho_{i,k} = \frac{m_{k,l}}{M_i} \quad i = 1, 2, \ldots
\]

\[
\sigma_{i,k} = \frac{m_{k,l}}{M_i} \quad i = 1, 2, \ldots
\]

We can determine the impact of \( f_k \) defined as the number of impacts multiplied by

\[
f_{a,k} = \frac{m_{k,l}}{M_i} \quad i = 1, 2, \ldots
\]

\[
f_k = \frac{m_{k,l}}{M_i} \quad i = 1, 2, \ldots
\]
described above includes \( m_{k,i} \). The momentum calculation is made to determine the normal velocity of master node \( l \) where the summation is performed over all of the slave nodes contributing to the mass of master node \( l \),

\[
v_{n_l}^+ = [(M_l^+ - m_{k,i}^+)v_{n_l}^- + m_{k,i}^+v_{n_l}^-]/M_l^+ ,
\]

\[
M_l^+ = \sum_i m_{l,i} + M_l = \Delta M_l + M_l ;
\]

\( M_l \) is the mass at node \( l \), \( v_{n_l}^- \) and \( v_{n_l}^- \) are the pre-impact velocities of master node \( l \) and slave node \( k \), respectively;

\[
v_{n_l}^- = \dot{z}_l^- \cos \theta_{l-1,L-1} - \dot{y}_l^- \sin \theta_{l-1,L-1}, \quad v_{n_k}^- = \dot{z}_k^- \cos \theta_{l-1,k+1} - \dot{y}_k^- \sin \theta_{l-1,k+1} .
\]

Defining the tangential component of the velocity with

\[
v_{t_l}^- = \dot{y}_l^- \cos \theta_{l-1,L-1} + \dot{z}_l^- \sin \theta_{l-1,L-1} ,
\]

the global post-impact velocities of \( l \) are given by

\[
y_l^+ = v_{t_l}^+ \cos \theta_{l-1,L-1} - v_{t_l}^- \sin \theta_{l-1,L-1} , \quad z_l^+ = v_{n_l}^+ \cos \theta_{l-1,k+1} - v_{n_l}^- \sin \theta_{l-1,k+1} .
\]

2.4.6. Master surface force distribution

The pressure acting on a master surface segment is interpolated from the slave elements in contact with the segment. This is accomplished with identical procedures as used to interpolate stresses for the phony elements in the TOODY scheme; therefore, (2.9)–(2.13) apply here as well. However, rather than needing to interpolate the entire stress tensor, bulk \( q \), and density, we need only interpolate the pressure distribution that equilibrates the normal components of force along the slideline.

With reference to Fig. 3, the contribution of pressure to master segment \( l \) made by slave element \( k \) is designated by \( p_{l,k} \) and defined by (2.36), where \( \sigma_l^k \) is the effective stress in slave element \( k \) perpendicular to master segment \( l \),

\[
p_{l,k} = (S_{l,k+1} - S_{l,k})\sigma_l^k / S_l ,
\]

\[
\sigma_l^k = \frac{f_{n_k} + f_{n_{k+1}}}{L_k} .
\]

We can determine the normal forces with (2.38), where \( f_{y_k} \), \( f_{z_k} \), \( f_{y_{k+1}} \) and \( f_{z_{k+1}} \) are components of \( f_k \) defined by the integral in (2.39) of the transpose of the strain displacement matrix \( B \) multiplied by the element stress vector \( \sigma \) over the current geometry,

\[
f_{n_k} = f_{z_k} \cos \theta_{l-1,k+1} - f_{y_k} \sin \theta_{l-1,k+1} , \quad f_{n_{k+1}} = f_{z_{k+1}} \cos \theta_{l-1,k+1} - f_{y_{k+1}} \sin \theta_{l-1,k+1}
\]

\[
f_k = \int_{V_k} B^t \sigma \, dV_k .
\]
If inequality (2.9d) is satisfied, \( p_{l,k} \) becomes the master segment pressure, otherwise the contribution to the pressure of the next overlapping slave element is computed:

\[
p_{l,k+1} = (S_{l,k+2} - S_{l,k+1}) \sigma_{l}^{k+1}/S_{l}.
\]  

(2.40)

If (2.11b) is not satisfied, \( k \) is incremented by 1 and the procedure given by (2.11) and (2.40) continues until (2.11b) is satisfied. The total pressure for master segment \( l \) is then given by

\[
p_{l} = \sum_{i=0}^{n-1} p_{l,k+i}
\]  

(2.41)

(the summation is performed over all of the slave segments in contact with the master segment).

The nodal forces acting at master node \( l \) due to the pressure distribution on master segments \( l-1 \) and \( l \) are

\[
f_{y_{l}} = \frac{1}{2} \left[ (z_{l} - z_{l-1}) p_{l-1} + (z_{l+1} - z_{l}) p_{l} \right],
\]

\[
f_{z_{l}} = -\frac{1}{2} \left[ (y_{l} - y_{l-1}) p_{l-1} + (y_{l+1} - y_{l}) p_{l} \right].
\]  

(2.42)

2.4.7 Interface forces and master surface acceleration

The interface force calculation is based on the TENSOR algorithm. At master node \( l \), the normal interface force, \( \Delta f_{n_{l}} \), is given by

\[
\Delta f_{n_{l}} = \frac{\Delta M_{l} F_{n_{l}} - M_{l} f_{n_{l}} + \Delta M_{l} f_{n_{l}} + M_{l} a_{e}}{M_{l}},
\]  

(2.43)

where

\[
F_{n_{l}} = M_{l}(\ddot{z}_{l} \cos \theta_{l-1,l+1} - \ddot{y}_{l} \sin \theta_{l-1,l+1}),
\]

\[
f_{n_{l}} = f_{z_{l}} \cos \theta_{l-1,l+1} - f_{y_{l}} \sin \theta_{l-1,l+1},
\]

and \( a_{e} \) is the Coriolis acceleration defined in (2.15). Normal and tangential accelerations of the \( l \)th master node are

\[
\ddot{a}_{n_{l}} = (F_{n_{l}} - \Delta f_{n_{l}})/M_{l}, \quad \ddot{a}_{t_{l}} = \dot{y}_{l} \cos \theta_{l-1,l+1} + \dot{z}_{l} \sin \theta_{l-1,l+1}.
\]  

(2.44)

The global accelerations can now be found from (2.6).

2.4.8 Slave node accelerations and velocities

The preceding steps have determined the motion of the master surface and have left unchanged the motion of the slave surface. As might be expected, the motion of slave nodes not in contact with the master surface is unaffected, but the motion of those slave nodes in contact with the master surface must be adjusted to ensure that the latter nodes remain so. In the procedure used here, the velocities and accelerations normal to the master line are reset.

Although new we set the master segment pressure to zero.

Consider the tangential accelerations of the master surface. The acceleration of node \( k \) is

\[
\ddot{v}_{k} = \ddot{v}_{k} + \ddot{v}_{k+1}.
\]

(2.45), wherein

\[
\dot{v}_{k} = \dot{v}_{k},
\]

We shall determine the tangential accelerations of the slave nodes and we shall interpolate for node \( k \)

\[
\ddot{v}_{k} = \ddot{v}_{k} + \ddot{v}_{k+1}.
\]

(2.46)

A normal acceleration of the master surface is

\[
\ddot{v}_{k} = \ddot{v}_{k} + \ddot{v}_{k+1}.
\]

(2.47), wherein

\[
\dot{v}_{k} = \dot{v}_{k},
\]

The new velocity of node \( k \) is

\[
y_{k} = y_{k} + \dot{y}_{k} t + \frac{1}{2} \ddot{y}_{k} t^{2}.
\]

Throughout this section we consider results of a large normal acceleration.

\[
\psi_{k} = \psi_{k} + \omega_{k} t + \frac{1}{2} \psi_{k} t^{2}.
\]

(2.48)

where \( \psi_{k} \) is the tangential acceleration of the slave node.
Although no guarantee exists in the present algorithm that slave nodes will remain exactly on the master surface, in practice, excursions away from the master surface have proven to be negligible.

Considering each slave node in turn, the first step is to compute the tangential velocity and acceleration. For the \( k \)th slave node on the \( j \)th master segment, these quantities are given by (2.45), where \( \psi_k \) measures the angle of the slideline at the location of node \( k \):

\[
v_{k,i} = \dot{y}_k^i \cos \psi_k + \dot{z}_k^i \sin \psi_k, \quad a_{k,i} = \ddot{y}_k^i \cos \psi_k + \ddot{z}_k^i \sin \psi_k.
\]

(2.45)

We shall discuss the choice of \( \psi_k \) later. The normal components of velocity and acceleration are interpolated from the master nodes. For example, the normal component of the velocity for node \( k \) sliding on master segment \( l \) is defined by

\[
v_{k,s} = (1 - \alpha_k) [\dot{z}_l \cos \psi_k - \dot{y}_l \sin \psi_k] + \alpha_k [\dot{z}_{l+1} \cos \psi_k - \dot{y}_{l+1} \sin \psi_k].
\]

(2.46)

A normal acceleration component, \( a_{k,n} \), is likewise defined. After the tangential and normal components of the acceleration and velocity are known, the new accelerations are given by (2.47), where the last term is the Coriolis term and \( \dot{y}_k^i, \dot{z}_k^i \) define the relative velocities between master segment \( l \) and node \( k \),

\[
\ddot{y}_k^{n+1} = a_{k,n} \cos \psi_k - a_{k,n} \sin \psi_k - 2\alpha \dot{z}_k^i,
\]

(2.47)

\[
\ddot{z}_k^{n+1} = a_{k,n} \sin \psi_k + a_{k,n} \cos \psi_k + 2\alpha \dot{y}_k^i,
\]

(2.48)

\[
\omega = (v_{n+1} - v_n)/S_l,
\]

where \( \omega \) is the angular velocity of the \( l \)th segment.

(2.49)

The new velocities are the by now familiar combination of the tangential components from the slave side and normal components from the master side,

\[
y_k = v_{k,i} \cos \psi_k - v_{k,s} \sin \psi_k, \quad z_k = v_{k,i} \sin \psi_k + v_{k,s} \cos \psi_k.
\]

(2.50)

Throughout the development of our slideline logic we have studied the choice of \( \psi_k \) on the results of a large number of applications. In our first implementation of the slideline logic we set

\[
\psi_k = \begin{cases} 
\theta_{l,i+1}, & \varepsilon \leq \alpha_k \leq 1 - \varepsilon, \\
\theta_{l-1,i+1}, & \alpha_k < \varepsilon, \\
\theta_{l,i+2}, & \alpha_k > 1 - \varepsilon,
\end{cases}
\]

(2.51)

where \( \varepsilon \) is typically set to \( 10^{-3} \). We found that this choice created numerical noise in some calculations, usually in the form of hourgassing, and that computing \( \psi_k \) as the normal to a quadratic curve through the closest master node and two adjacent nodes eliminated much of the noise. If \( l \) is the closest master node, then (2.52) is the parametric equation for a parabola

\[
\psi_k = \theta_{l,i+1}, \quad \varepsilon \leq \alpha_k \leq 1 - \varepsilon,
\]

(2.52)
where the parametric coordinate $\xi$ is bounded in the interval from $-1$ to 1 inclusive. Angle $\alpha$ is now based on the slope of (2.52):

$$y(\xi) = \frac{1}{2} \xi (\xi - 1) y_{l-1} + (1 - \xi^2) y_l + \frac{1}{2} \xi (\xi + 1) y_{l+1},$$

$$z(\xi) = \frac{1}{2} \xi (\xi - 1) z_{l-1} + (1 - \xi^2) z_l + \frac{1}{2} \xi (\xi + 1) z_{l+1};$$

$$\psi_k = \cos^{-1}\left\{\left(\frac{\frac{dy}{d\xi}}{\left(\frac{dy}{d\xi}\right)^2 + \left(\frac{dz}{d\xi}\right)^2}\right)^{1/2}\right\}_{\xi = \alpha_k}.$$ (2.53)

We assume that node $k$ is sliding on master segment $l$. If $l + 1$ is the closest node to $k$, the foregoing procedure (2.52)-(2.53) is repeated with $l$ increased by 1 and $\xi$ is $\alpha_k - 1$. Equations (2.52) and (2.53) are used in the public domain versions of DYNA2D.

Using (2.53) to determine $\psi_k$ leads to the same kind of noise as (2.51) though not as pronounced. We can illustrate this noise with the concentric spheres in Fig. 8 which have the bulk modulus of steel and no shear strength. The spheres are 1 centimeter thick and 7.5 and 8.5 centimeters in radius. A 100 kilobar pressure is applied to the outer surface to push the spheres inward.

![Initial geometry of concentric spheres. A 100 kbar pressure is applied to the outer surface to push the spheres inward.](image)
Angle $\psi_k$

\[
\psi_k = \theta_{k-1,k+1},
\]

we get the more pleasing results in Fig. 10.

We can expect separation between the master and slave surfaces if normals are based on (2.54). In many of our calculations we have found the separation or inner penetration to be very small, but in other calculations the separation or overlap is noticeable. In these latter cases very slight adjustment to the normal slave velocity is all that is necessary to keep the slave nodes close to the surface. We now use the normal distance from the slave node to the master surface to compute the velocity change required to close the gap, and we adjust the normal slave velocity each time step by 1 percent of this change. Thus, the slave node is never returned to the master surface; it is just kept reasonably close. We have not found momentum conservation to be significantly affected and, in fact, it has sometimes been better than with our old approach.

Fig. 9. Improper interface normal leads to anomalous behavior.
2.4.9. Time-step calculation

Before updating the configuration of $t^{n+1}$, every slave node that is not in contact with the master line is checked to determine if it will penetrate the master line between time $t^n$ and $t^{n+1}$. If any slave node penetrates, $\Delta t^{n+1/2}$ is scaled back such that no slave node penetrates by at least one slave node reaches the master surface. At the beginning of the next time step, impact conditions are applied to the slave nodes that reached the master line and constraints are imposed to keep the nodes in contact with the master line.

Consider slave node $k$ that is close to but not in contact with the master line at time $t^n$. The first step in determining penetration is to locally update the geometry of node $k$ and nearby master nodes. After the closest master node is located, the master segment associated with slave node $k$ is found using (2.18) and (2.19). Let master segment $l$ be associated with slave node $k$. If (2.55) is satisfied, where $m_l$ is the outward drawn unit vector normal to master segment $l$, penetration has occurred:

$$ r_{kl} \cdot m_l < 0. $$

If slave node $k$ penetrates master segment $l$, the maximum permissible time-step size is

$$ F_w \text{ and } F_s \text{ are the force and stiffness added due to the } $
computed that will not allow node $k$ to penetrate. This time-step size is found by equating the slope of master segment $l$ with the line drawn from node $l$ to $k$ at $t^n + \Delta t$. If the velocities are assumed to be constant, a quadratic equation is obtained for $\Delta t$.

2.4.10. Simplifications

We have attempted to increase the efficiency of the foregoing algorithm by making two reasonable approximations. In the first we replace the calculation for $m_k$ in (2.25) with an approximation using the mass matrix

$$m_k = \frac{1}{2}L_k(m_k^* + m_{k+1}^*), \quad m_k^* = 2M_k / |r_{k-1,k+1}|;$$

(2.56)

$m_k$ is the global mass at slave node $k$, and $|r_{k-1,k+1}|$ denotes the length of the vector from $k - 1$ to $k + 1$. In the second approximation we replace calculations to obtain $\sigma_k^s$ in (2.37) by

$$\sigma_k^s = \frac{\frac{1}{2}(\sigma_k^* + \sigma_{k+1}^*)}{|r_{k-1,k+1}|}, \quad \sigma_k^* = \frac{2(F_{zk} \cos \theta_{k-1,k+1} - F_{zk} \sin \theta_{k-1,k+1})}{|r_{k-1,k+1}|};$$

(2.57)

$F_{zk}$ and $F_{zk}$ are the global forces at slave node $k$ before accounting for the slideline coupling.

We have vectorized the slideline logic that includes these simplifications and, consequently, have reduced the cost significantly for what seems to be only a minor difference in the results due to the averaging.

2.4.11. Extension of hydrocode algorithm to 3D

We have implemented the sliding interface algorithm without separation between the surfaces into DYNA3D. Because we know of no reasonable way to economically map the slave surface on to the master surface for the distribution of mass and pressure, the simplifications of Section 2.4.10 are used in the 3D extension.

3. Penalty formulation

The computer implementation of the penalty formulation in explicit codes is straightforward. The interface subroutines are called each time step prior to computing the accelerations. In the implicit codes the interfaces are treated like another element class. We reform the stiffness matrix at the beginning of each time step taking into account the changing connectivity along the interfaces; consequently, we do not limit the relative displacement. We also reperform during the equilibrium iterations if convergence problems arise. Bandwidth minimization is performed at the beginning of the problem and is based on the initial coupling across the interface.

In two dimensions, the slideline definitions are identical to the DYNA2D hydrocode algorithm. In three dimensions, all triangular and quadrilateral segments in each surface are required as input with an arbitrary ordering of the segments. In the development that follows, we refer to one surface as the slave and the other as the master even though the symmetry of our approach eliminates any bias in this choice.

Initialization of the penalty method in two dimensions involves several steps. First, we
reverse the ordering of the nodes along the slave side so that as one moves along the slave
surface in the order that the slave nodes appear, the master surface will be to the left, and then
we check each master and slave node for penetration through the slave and master surfaces
respectively. Penetrating nodes are projected back on to the surface. Finally, for each master
and slave segment, we find the element that contains the segment in its connectivity and
compute a segment stiffness as a function of the bulk modulus and thickness. The second step
eliminates the possibility of large forces developing due to interpenetration of nodes along
the curved interface due to discretization differences across the interface, and the third step
ensures a reasonable value for the interface stiffness.

In the three-dimensional initialization we first gather all the slave and master nodes into
dd arrays and store the segment numbers that contain each node. The second and third steps are
the same.

A general interface treatment using penalty functions may be outlined as follows:

Step 1. For each slave node, \( n_s \), locate the closest master node, \( n_m \), and check the master
segments that include \( n_m \) to identify the segment, if any, containing \( n_s \).

Step 2. Locate the position of the slave node on the master surface.

Step 3. Determine if \( n_s \) has penetrated the master segment. If so, compute and add an
interface force to the right-hand side and, if the global stiffness matrix is being reformed, add
in an interface stiffness matrix.

Step 4. Repeat Steps 1–3 for the master nodes.

3.1. Determination of master segment containing slave node

Consider a slave node, \( n_s \), sliding on a piecewise smooth master surface and assume that
search of the master surface has located and stored the master node, \( n_m \), lying closest to \( n_s \).
Fig. 11 depicts a portion of a master surface with nodes \( n_m \) and \( n_s \) labeled. To minimize the
operation count, the search for the closest node only includes the closest node from the
previous time step, \( n_m^{old} \) and its surrounding nodes which are available in the connectivities of
the segments that contain \( n_m^{old} \).

If \( m_s \) and \( n_s \) do not coincide, \( n_s \) can usually be shown to lie in a segment \( s_i \) via the following
tests:

\[
(c_i \times s) \cdot (c_i \times c_{i+1}) > 0 , \quad (c_i \times s) \cdot (s \times c_{i+1}) > 0 .
\]  (3.1)

Vectors \( c_i \) and \( c_{i+1} \) are along edges of \( s_i \) and point outward from \( m_s \), and vector \( s \) is the
projection of the vector beginning at \( m_s \), ending at \( n_s \), and denoted by \( g \), on to the plane being
examined (see Fig. 12):

\[
s = g - (g \cdot m) m , \quad m = \frac{c_i \times c_{i+1}}{|c_i \times c_{i+1}|} .
\]  (3.2)

Since the sliding constraints keep \( n_s \) close but not necessarily on the master surface and since
\( n_s \) may lie near or even on the intersection of two master segments, the inequalities of (3.1)
may be inconclusive, i.e., they may fail to be satisfied or more than one may give positive
results. When this occurs \( n_s \) is assumed to lie along the intersection which yields the maximum
Assume that $n_s$ is the slave closest to $n_i$. To minimize the distance from the master velocities of $n_s$, the following conditions must hold:

$$s \cdot c_i = 0, \quad c_i \geq 0$$

(3.1)

where $s$ is the vector from $n_s$ to $n_i$, and since $c_i$ is a unit vector, the conditions of (3.1) are equivalent to maximizing $s \cdot c_i$. The value for the quantity

$$\frac{g \cdot c_i}{|c_i|}, \quad i = 1, 2, 3, 4, \ldots$$

(3.3)
The test (3.1) is first applied to the master segment that contained the slave node in the previous step. If (3.1) is not satisfied, then each master segment that includes \( n_m \) is checked. Our implementation does not limit the number of segments that may contain \( n_m \).

### 3.2. Determination of the contact point

Assume that a master segment has been located for slave node, \( n_s \), and that \( n_s \) is not identified as lying on the intersection of two master segments. Then we may identify the 'contact point', defined as the point on master segment which is closest to \( n_s \). Each master surface segment, \( s_n \), has a bilinear parametric representation,

\[
 r = f_1(\xi, \eta)i_1 + f_2(\xi, \eta)i_2 + f_3(\xi, \eta)i_3 ,
\]
\[
f_i(\xi, \eta) = \sum_{j=1}^{4} \phi_j x^j , \quad \phi_j(\xi, \eta) = \frac{1}{2}(1 + \xi_j)(1 + \eta_j); \tag{3.5}
\]

\( \xi_j, \eta_j \) take on their nodal values at \((\pm 1, \pm 1)\), and \( x^j \) is the nodal coordinate of the \( j \)th node in the \( j \)th direction, see Fig. 13. Note that \( r \) is at least once continuously differential and that the normal is nonzero,

\[
 \frac{\partial r}{\partial \xi} \times \frac{\partial r}{\partial \eta} \neq 0. \tag{3.6}
\]

Fig. 13. Location of contact point when \( n_s \) lies above master segment is normal to the master segment at the contact point.

Thus, \( r \) represents a local coordinate system.

Let \( t \) be the unit vector along the segment \( s_n \),

\[
t = \frac{\partial r}{\partial \xi} \times \frac{\partial r}{\partial \eta} .
\]

The physical equation to be satisfied is to solve for \( t \) in the equation

\[
f \cdot t = 0 .
\]

### 3.3. Nodal penetration

Each slave node is checked to determine if it lies within the master segment and if the penetration is zero. Penetration is the distance from the slave node to its projection on the master segment. Thus, a slave node \( n \) is considered to lie in contact if

\[
f_1(n \in s_n) = 1 .
\]

If slave node \( n \) lies outside the master segment, then \( f_1(n \in s_n) = 0 \) and an equal penetration distance from the contact point is calculated.
Thus, $r$ represents a master segment that has a unique normal whose direction depends continuously on the points of $s_n$.

Let $t$ be a position vector drawn to slave node $n_s$ and assume that the master surface segment $s_i$ has been identified with $n_s$. The contact point coordinates $(\xi_c, \eta_c)$ on $s_i$ must satisfy

\[
\frac{\partial r}{\partial \xi} (\xi_c, \eta_c) \cdot [t - r(\xi_c, \eta_c)] = 0 , \tag{3.7a}
\]

\[
\frac{\partial r}{\partial \eta} (\xi_c, \eta_c) \cdot [t - r(\xi_c, \eta_c)] = 0 . \tag{3.7b}
\]

The physical problem is illustrated in Fig. 13 which shows $n_s$ lying above the master surface. Equations (3.7a) and (3.7b) are readily solved for $\xi_c$ in terms of $\eta_c$. One way to accomplish this is to solve (3.7a) for $\xi_c$ in terms of $\eta_c$, and substitute the result into (3.7b). This yields an equation in $\eta_c$ which is presently solved numerically in DYNA3D.

### 3.3. Nodal force update

Each slave node is checked for penetration through its master segment. If the slave node does not penetrate, nothing is done, but if it does, an interface force is applied between the slave node and its contact point. The magnitude of this force is proportional to the amount of penetration.

Penetration of the slave node $n_s$ through the master segment $s_i$, which contains its contact point is indicated if $l$ is negative,

\[
l = n_i \cdot [t - r(\xi_c, \eta_c)] < 0 , \quad n_i = n_i(\xi_c, \eta_c) = \frac{\frac{\partial r}{\partial \xi} \times \frac{\partial r}{\partial \eta}}{\left| \frac{\partial r}{\partial \xi} \times \frac{\partial r}{\partial \eta} \right|} . \tag{3.8}
\]

If slave node $n_s$ has penetrated through master segment $s_i$, we add an interface force vector $f_s$ that is normal to the master segment and linearly dependent on the penetration of node $n_s$,

\[
f_s = -lk_i n_i \quad \text{if} \quad l < 0 . \tag{3.9}
\]

An equal and opposite force is distributed over the master segment nodes according to

\[
f'_m = -\phi_i(\xi_c, \eta_c)f_s \quad \text{if} \quad l < 0 . \tag{3.10}
\]

The stiffness factor $k_i$ for master segment $s_i$ is given in terms of the bulk modulus, $K$, the volume $V$, and the face area, $A_i$, of the element that contains $s_i$ as

\[
k_i = \frac{f_{si} K_i A_i^2}{V_i} , \tag{3.11}
\]

where $f_{si}$ is a scale factor for the interface stiffness and is normally defaulted to 0.1. Larger values may cause instabilities unless the time-step size is scaled back in the time-step calculations.
4. Examples

We present in this section five examples from diverse applications where contact-impact algorithms are necessary. None of these examples were run just for this paper; they were run on the production versions of our codes in the production-oriented engineering environment at Lawrence Livermore National Laboratory.

EXAMPLE 4.1. Mass focused projectiles. Our first example demonstrates the hydrodynamic approach to contact implemented in DYNA3D.

Mass focused projectiles, also called Misznay–Schardin munitions, are being designed as anti-armor devices. This kind of munition uses high explosives to accelerate a metal plate to a high velocity. As the plate accelerates, it is designed to deform into a cohesive shape that can penetrate thick armor. Furthermore, its shape must provide aerodynamic stability to accommodate long standoffs from the target. Computer codes are used to iteratively develop a design with all the desired properties. Testing of the design follows to insure that everything performs as predicted.

Tuft and Godfrey [16] have computed fully three-dimensional geometries such as shown in Figs. 14 and 15 with two planes of symmetry. The time sequence of two liner cross sections is shown in Fig. 16.

![Fig. 14. Basic geometry for 3D mass focused projectile device.](image-url)
Slidelines are used between the high explosive, the liner and the case to allow the gas from the explosion to expand without restriction. The deformed shape of the liner is very sensitive to the pressure history, therefore both the detonation wave front and explosive-liner interface must be modeled carefully.

**EXAMPLE 4.2. Bellows forming analysis.** This example illustrates a problem involving a large amount of sliding along the contact interface. The purpose of the calculation is to determine the pressure required to collapse the stainless steel sleeve into the elastic die. Fig. 17 depicts the calculational mesh and boundary conditions.

A pressure loading is applied in 43 equal increments to a peak value of 0.580 GPa and is removed in two increments for a total of 45 steps. When the pressure reaches 0.483 GPa, the sleeve is completely collapsed. A sequence of deformed shapes is shown in Fig. 18 followed by the final configuration in Fig. 19.

The analysis was performed using NIKE2D; the slideline uses the penalty formulation.

**EXAMPLE 4.3. Metallic O-ring.** In a recent application, standard metallic O-rings were failing to produce a reliable seal. The designers felt that these failures were due to large cutouts in one of the flanges. It was felt that these cutouts, which are required for the
installation gas transfer fittings, reduced the rigidity of the flange to the point where the interface pressures were insufficient for sealing.

The following figures show the three-dimensional model which was used to analyze the problem. The mesh in Figs. 20 and 21 consists of 4816 nodal points and 3304 elements. Slidelines are defined between the upper portions of the O-ring and the upper flange, the lower portion of the O-ring and the lower flange, and the outer portion of the O-ring and the corresponding sealing surface of the lower flange. Additionally, a slide-surface is located between the two flanges. The model extends from the center of one of the cutouts to the center of a corresponding bolt hole. Symmetry planes are used at each of these locations to provide boundary conditions.

The total loading of the model occurs over fifteen steps. The first four steps crush the O-ring. During this portion of the loading, the O-ring undergoes large plastic strain and the interface pressures begin to develop. The next two steps are used to finish closing the gap between the two flanges. Steps eight through ten finish the bolt loading of the flanges. The upper bolt loads are sufficient to cause some plastic deformation of the upper flange. The last five steps are used to apply the gas pressure to the inside of the model. This pressurization partially unloads the O-ring. Fig. 22 shows a result of the calculation.

The NIKE3D analysis clearly showed that the cutouts were not causing the failures.
Fig. 17. Initial configuration.

Fig. 18. Deformed shapes corresponding to pressures of (a) 0.138 GPa, (b) 0.276 GPa, and (d) 0.552 GPa.
**EXAMPLE 4.4. Pipe whip.** Our final penalty formulation example uses the recently implemented shell elements in NIKE3D.

Current Nuclear Power Plant design regulations require pipes to be designed against pipe whip. It is postulated that in the event of pipe break, the high pressure fluid in the pipe could cause the pipe to swing and impact another pipe. The Nuclear Regulatory Commission has provided multi-year funding to Livermore to study this problem.

A sample problem was developed to test the ability of NIKE3D to solve pipe impact problems. The problem consists of two steel pipes both having a thickness of 0.432 inches, a length of 33.125, a length of 50 inches. One pipe is oriented horizontally and completely fixed at both ends, and a yield strength of 10,000 psi, and a hardening modulus of 10^5 psi. The other pipe swings about one end in a plane normal to the horizontal pipe. The angular velocity at the time of impact is 50 radians/sec. Shell elements were used in the mesh with one plane of symmetry. Two-hundred steps were used with a termination time of 10 ms. Results from the pipe impact are shown in Figs. 23-25. The pipe begins to rebound at approximately 7 ms.

Several experiments on pipe impacts were performed for the Nuclear Regulatory Comm...
Fig. 21. Close-up view of O-ring at start of calculation.

Fig. 22. Deformed O-ring at the completion of the sixth time step.
Fig. 23. Sequence of deformed shapes in 2 ms intervals.

Fig. 24. Rotated view of final configuration.
mission that could provide a basis for verifying the shell impact capability. Efforts were made to obtain the experimental data, but were only obtained after this paper went to press.

References