Three-Dimensional Surface Contact

by

Samuel McDonald
MDI Professional Services
2301 Commonwealth Blvd.
Ann Arbor, Michigan 48105 U.S.A.
313-994-3800

ABSTRACT:

In mechanical system simultaion, there is often a need to model the surface contact of one body to another. In ADAMS, every contact must be modeled by the user, which can make some types of analysis difficult, if not impossible for the ADAMS novice. This paper will specifically outline a procedure for creating a sphere to generalized surface contact that can be easily modeled by the casual user, while showing a general method for modeling 3-D contact. (The subroutine and macros described by this paper are on the 8.0 distribution CD underneath the aview/examples/contact directory).

OVERVIEW OF MODELING PROCEDURE:

To create the surface contact, the user creates a simple 3-d geometry in his model. Under the menus for geometry, the user converts the geometry into a contact surface. Then the user creates a part for the sphere. To create the force, select the menu choice SURFACE from the element -like forces. Assuming geometry has been added, the model is now ready to run.

Build the model, submit the analysis, and post-process a 3-d contact in less than 15 minutes.

TECHNICAL OVERVIEW:

In this method of modeling, 3-D contacts were modeled by first finding the contact point information, and then passing that information to a general subroutine that calculates the six components of the force as applied to the I marker of the GFORCE. There is no need to have a marker at the contact point, as the general subroutine will calculate the equivalent forces and resultant torque's arising from the surface normal force, as well as resolve the forces due to sliding and 'static' friction. The shape of the I body need not be restricted to a sphere, but can be any shape desired as long as the user can supply the contact point information.

The information needed about the contact point includes the location of the contact, the normal of the surface at the contact point, the amount of penetration of the I surface into the J surface, the velocities of the two parts, and the ADAMS/IMPACT information detailing the type of contact. (e.g. SK, E, etc.) In the example on the CD, the IMPACT information is passed down via the parameter list for the GFORCE, while the contact information is determined inside the SPHSHL3.f subroutine (which is called from the GFOSUB)

There are three main components apparent to the user modeling surface contact in this way:

1. Surface representation.

2. Contact parameters.

3. Force representation.

1. Surface representation.

SHELL

The shell geometry in ADAMS/View consists of a set of polygons, usually (though not always) four sided. A shell cannot be directly created in ADAMS/VIEW, but must be read in from a file. The file consists of a series of points that are the vertices of the polygons, and a set of connectivity information that links together certain point for each individual polygon (from now on referred to as element). The shell in ADAMS/View is located in the model by specifying a reference marker, which is used as the reference frame origin by the points in the files.

The method here uses these elements as planes for the sphere to hit. The sphere can contact the center of the element, (resulting in a sphere-plane type contact,) or the edge of the element (resulting in either an sphere-line type contact or sphere-point type contact).

A shell can be created in many different ways. One is to use the AVIEW command GEOMETRY -> CREATE -> SHAPE -> SHELL to read in an already existing shell file.

Another way is to use the ADAMS/IGES translator and read in a .igs file. If the CAD model that the IGES file comes from uses a trimmed surface representation of the desired surface, then when it is transferred to AVIEW, it will come across as a SHELL geometry. The file can then be written out by writing a command file of the dataset, which will cause a .shl file to also be created.

A third way is to use a GEOMETRY from AVIEW, and convert it into a shell using the macro under the menu string GEOMETRY -> CREATE -> SHAPE -> ShellCon
This will convert the geometry by writing out a wavefront .obj file, then reading the .obj file back in. This creates a shell out of whatever geometry that was on the original part.

In all these methods, care must be taken to ensure that all the elements only have three or four sided elements in the shell file. The subroutine does not handle other than that.

2. Contact information

The contact information is very similar to that used for the ADAMS IMPACT function. The user will need to determine the following parameters:

- Free length (radius of sphere)
- Stiffness of contact
- Exponent of Displacement
- Vertical Damping Coefficient
- Penetration depth for full damping
- Coefficient of coulomb friction (static)
- Velocity for static friction measurement
- coefficient of sliding friction (sliding)
- Velocity for sliding friction measurement

These IMPACT parameters can be determined as would normal IMPACT parameters.

3. Force representation

Using the Macro under the ADAMS/View menu selection FORCE -> CREATE -> ELEMENT -> SURFACE will let you create a GFORCE statement that will call the sphere to shell subroutine. The macro will ask for the name of the ellipsoid that you want to represent the sphere, and the shell that you want to use for the surface. Optionally, you can also define the previously listed parameters, or use the defaults.

SUBROUTINES:

SPHSHL3.f

The subroutine will check every element in the shell to determine if it is contacting with the sphere. The contact information comes out naturally as a consequence of this search. Before the elements get checked for contact, each element in the shell gets its normal (NORM) and area (AREA) calculated and stored. (when IFLAG is true)

The penetration depth is calculated (D in the SPHSHL subroutine) by finding the distance between the center of the sphere, and the plane with the equation:

D = NORMX * (DX-X1) + NORMY*(DY-Y1) + NORMZ*(DZ-Z1)

where X1,Y1,Z1 is a point in the plane, and DX,DY,DZ is the location of the center of the sphere with respect to the RM of the shell, in the coordinate system of the shell. This formula can lead to some difficulties if the sign of the normal is not known ahead of time. The lines immediately following the determination of D in the subroutine help prevent the sphere from starting off buried deep within the shell. This allows the simulation to start, but can lead to the sphere passing through the surface if the sphere radius is small enough and the integrator time step large enough.

Next, if D (which is the distance from the sphere center to the plane) is larger than the radius of the sphere, then there can be no contact, and that element is discarded in the search. If D is smaller, then the type of possible contact must be determined. When the center of the sphere is projected along the negative of the element normal and is inside the polygon, the type of contact is a sphere contacting a plane. When the projected center lies outside the polygon, then it needs to be further checked to determine if the contact is a sphere-line contact, a sphere-point contact, or even no contact at all.

After the type of contact gets determined, the location of the point of contact is calculated, and from that information a vector is created from the I marker to the contact point, in the coordinate system of the shell's reference marker. This vector will be used to determine what torques should be applied at the I marker (in addition to the translational forces) to allow the actual point of application to be moved. In the subroutines it is stored as CONPOS (contact position).

When the contact position is known, then the normal is calculated (CNORM in the subroutine) if the contact is not sphere to plane, and the information is passed to the XXXXX1 subroutine. The contact forces and torques are then passed back to the sphshl subroutine, and added to the forces already determined during this iteration.

OTHER USES:

This technique is being used in some current projects. The following are some cases where the subroutines here might prove useful.

XXXXX1 (or a successor) can be used not only to model sphere to shell contact as here, but can also be used in almost any application where there is a need to apply a force at a location that is not constant on either of the two bodies being affected. This would allow simpler modeling of Gears with flexible shafts, and/or strange variances in their CV locations that may preclude kinematic modeling.

With a large enough radius, a sphere can be used to approximate a flat plane. Thus, with small modification to XXXXX1, Shell to 'plane' modeling can also be done easily.

BENEFITS:

1. 3-D surface contact with "static" and sliding friction.

2. True WYSIWYG Surface definition.

3. Complex shape definition with no apparent markers.
   (1 GFORCE, 3 MARKERS)

4. Allows more than one contact between sphere and surface during iterations.

CAUTIONS:

1. XXXXX1 assumes J body does not rotate.

2. Can be slow when there are many elements.
   (resolved with later versions)

3. If integrator step size is too large, sphere can pass through surface.