Automatic Contact Force Application Techniques

Sam McDonald
Senior Design Engineer, TRW

Todd DePauw
MDI

International ADAMS User's Conference
Ann Arbor, Michigan
June 10, 1998

Abstract

For many years, the most challenging part of modeling dynamics models in ADAMS has been writing force functions by hand. Many times, these forces were required to model two bodies contacting each other. Many times, simplifications had to be made, or function expressions debugged in the dataset, to allow even the simplest and most often used contacts to be considered for use in a model. ADAMS V9.1 takes a step towards ease of use with the addition of a small subset of surface contact pairs that are easy to define, and simple to use. This paper provides a history of how contact methodologies up to and including the built-in contact functionality. A series of example applications are included to show how the functionality can be combined in order to model more complex cases.

Review of Contact History

DRAM

There have been many ways to model contact between two parts that have been used in virtual prototyping models. DRAM, a predecessor to ADAMS, had the Simple Positioned, and the Inclined Position Force. These forces required subroutines to determine such things as contact positions and surface normals.

Early ADAMS Modeling

In ADAMS, however, these forces were not implemented. To model contacts in early versions of ADAMS (pre 6.0), it was often necessary to use 12 action-only SFORCES to model a general type of contact. This allowed for not only the translational and rotational action forces on the I part, but also six more translational and rotational forces for the J part. Because of this, Contacts (and some other forces) were very slow compared to what they could be. Many informed users writing contacts during this time used the Hoban accelerator in their subroutines, a subroutine that allowed the integrator to significantly speed execution times by eliminating redundant differencing for the required 12 forces.

Progress

For ADAMS 6.0 and after two new modeling objects were introduced which helped address the contact problem. These were:

1. Curve-curve constraint with GFORCE based impact, and

2. GFORCE in the dataset.

GFORCES incorporated the lessons learned with the Hoban accelerator, and eliminated the need for the 12 action-only SFORCES. The problem with the SFORCES and the definition of their function expressions in the dataset were, however, still there with the GFORCE only method. Any non-trivial type of contact practically required either a subroutine, or 6 function expressions that could easily require more than 120 lines of expressions.

CV-CV Constraint

This new functionality allowed many engineers to attempt to ADAMS Version 6.1 to solve contact related problems. This was possible primarily due to the new ease of use enabled by the GFORCE. There was the method developed by Kelly Londry that allowed the point-curve (or curve-curve) constraints to separate, and model ‘cam liftoff’. This method used the constraints with 'dummy' parts that would track the closest approach, and then apply an impact between the two dummy parts. If the forces that arose from the dummy parts shifting could be compensated for, the method would give good answers. However it could run into difficulties because of the constraints, the compensating applied forces, and the actual force setup.

CONPAMM

After Kelly Londry, Yurom Kirson developed a set of 2-D contact routines called CONPAMM. These included curve-flat, and curve-circle cam followers, as well as some other types of 2-D contacts. These contacts worked well in the cases where they were used, but further MDI development stopped when Yurom’s sabbatical was over and he left MDI. The routines were delivered to MDI, but have only been used several years ago in a few MDI consulting projects.

ADAMS/Android

ADAMS/Android allowed an ellipsoid to make contact with a finite plane. The contact used ADAMS forces to model the contacts, and allowed the ellipsoid to roll of the edge of the finite plane. This allowed the android model to (for example) sit on a seat, or have his feet actually contact the ground.

Sphere-Shell

In ADAMS 8.0, another method was introduced. It modeled contact between a sphere and a shell with a single GFORCE in a user-written subroutine. Friction was correctly modeled as long as the shell part did not move. This method was made available to the user community on the distribution CD ROM, complete with macros to assist in defining the GFORCE, as well as FORTRAN source code.

Virtual Knee

Shaun McGuan, of MDI Consulting, built upon these routines, and wrote the Virtual Knee toolkit, which would allow a human knee to be modeled in ADAMS. Virtual Knee results have correlated well.

Contact Toolkit

ADAMS 9.0 saw the launch of the contact toolkits, with the 3D#1 version available through MDI’s Consulting Department. It included the following types of contacts:

- sphere-sphere
- sphere-pulley
- sphere-shell
- shell-plane

These have been successfully used in many MDI consulting and customer projects including belts and gears.

Automatic Contact

ADAMS V9.1 includes contacts based on the technology in the contact toolkits. The first series of contacts that will be included are:

- Sphere - sphere (internal / external)
- Sphere - infinite plane
- Circle - infinite plane

These three types of contacts can also be used to model circle-circle (planar only) contact. The sphere-sphere contact force is used, but if the parts are constrained in plane then they can be used as circles. In ADAMS/View, circle geometry is used to create and parameterize the contact forces, but the user must ensure that the circles act in plane.

The combinations were chosen for their relative robustness to make sure that the method is suitable for including contact without causing undo modeling difficulties. The next section discusses some examples that illustrate the application of the available methods.

Revolute Joint (Slop and Friction)

The first example application is to use the circle-circle (inside) to simulate a revolute joint with friction and slop. To create this in a model, simply create two circles on the respective parts. Then use the circle geometry objects to create a contact force. The contact array values should be prescribed for the material and surface properties. The parts should be modeled to move in the same plane so that the circles will not rotate out of plane or translate in the direction normal to that plane. Otherwise the appropriate joint (such as a planar joint) can be added.

{ EMBED Word.Picture.8 }*Figure 1: Contact Based Revolute Joint*

Spherical Joint (Slop and Friction)

The same method can be applied to simulating spherical joints with slop and friction. In this case, the sphere-sphere (inside) force is created. The contact forces can also be defined using design variables, so design studies can be conducted based on the tolerance level, for example.

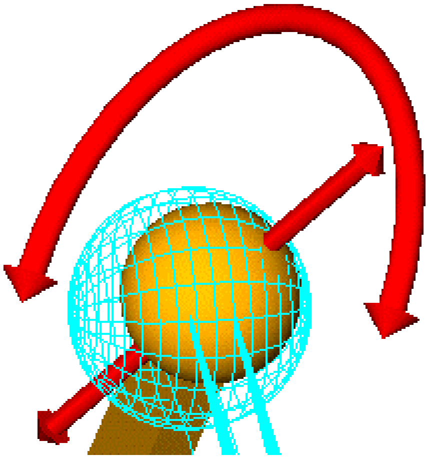

Figure 2: Contact Based Spherical Joint

Sphere-Cylinder

Although sphere to cylinder contact is not currently available, there is a method which can be used to simulate the combination. In this method, an extra part is required. A circle with the same radius as the cylinder is created and a translational joint between the extra part and the cylinder part allows the cylinder to translate along the cylinder axis. Then an inplane joint primitive is created with the I marker on the sphere and the J marker on the circle part. Finally a circle with the same radius as the sphere is required on the sphere's part. After this has been set up, a circle-circle (inside) contact force can be created. During the simulation, the constraints on the circle part will force it to stay in 2D planar space with respect to the sphere. Then if a contact occurs, the circle on the cylinder will be in the correct position. This method should be used with care. The additional circle parts should be created with extremely small masses and inertia in order to reduce the effect on system dynamics. Additionally, care should be used if the initial velocities of unconstrained parts have a large effect on system dynamics, then the method may cause undesirable velocities. Finally, care should be used to make sure the constraints are placed at the centers of gravity of the parts to ensure that the reaction forces for the extra parts do not cause inappropriate moments.

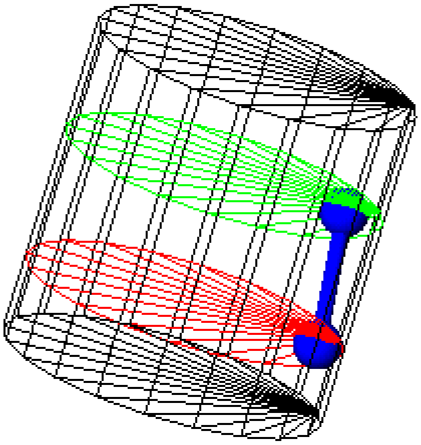

Figure 3: Satellite

Contact on a Discrete Flexible Link

The discrete flexible link (otherwise known as the beam macro) introduced in ADAMS 9.0 provides the ability to readily create a series of parts connected by beam forces. The contact forces can be used by adding commands to the beam macro to automatically create a circle object and a contact force for each segment. This method allows for modeling flexible connections that may come into contact with other objects during the simulation. Two examples of this application can be observed in flexible hoses in an engine compartment, or a vehicle exhaust system.

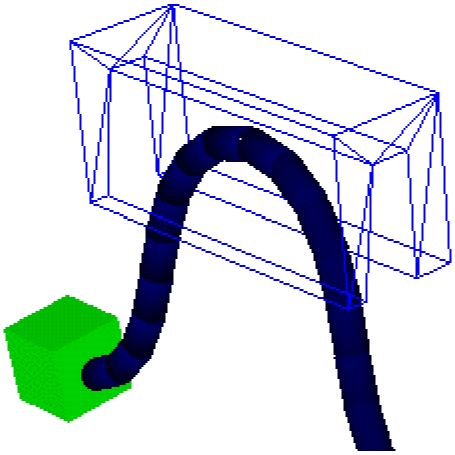

Figure 4: Contact on Multipart Beam

Contact and ADAMS/Flex

Naturally any place that a force is applied to a flexible body is a candidate for using the contact force functionality. As of the writing, the method has been applied by connecting a small part at the node location with a fixed joint, then applying the contact force to the part. For example, contact forces can be created on the exterior nodes of a flexible piston head. In this case, sphere-plane, can be used with a marker that allows contact with the plane tangent to the inside of the piston sleeve.

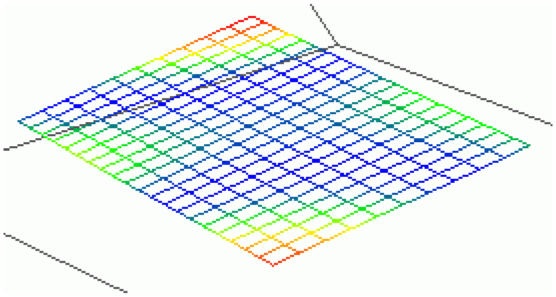

Figure 4: Flex-Body Contact

Simple Tires with Contact Forces

There are some cases where only a simple tire like force is required. For example, a medical cart’s stability can be simulated using contact forces, where traditional tire forces may be too detailed, or appropriate tire property files are not available. Also, the contact forces can be used to develop a model and do preliminary testing until including the traditional ADAMS/Tire forces become appropriate. To model simple tires with contact forces, simply create a circle on the tire part and a marker to define the plane for the tire to roll on. There are also some cases where standard ADAMS/Tire forces are not available. An example of this would be the need to model a vehicle that operates on the deck of a ship. Since ADAMS/Tire forces must react to ground, the ship motion can not be modeled. The contact forces allow the force to be correctly modeled on parts that can translate and rotate in space.

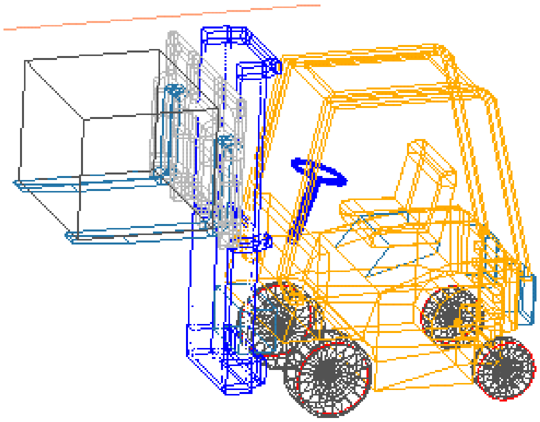

Figure 5: Simple Tire Contact Forces

Future Work

Since no work is underway at MDI to include additional contact pairs, the bulk of the effort in contact modeling will be to continue to gain experience in using the available pairs. As future methods of combining and applying the available pairs will be tracked and shared with the engineering community. In addition, it is understood that MDI’s product development department will continue to investigate appropriate methods of general contact to incorporate into future releases. Additionally, the contact toolkit available from MDI Consulting services provide additional pairs and additional pairs may be developed, as required, to complete future consulting projects and may be added to the commercially available toolkit.

Investigation is currently underway to determine the feasibility and market demand for providing additional contact pairs by independent contractors. If a business case can be made for continuing the work by providing a service to the engineering company, then that service may be made available. All questions regarding contact updates on new contact functionality are welcome.

Filename:	pap_contact.doc
Directory:	\\Davinci\UserConf
Template:	C:\Program Files\Microsoft Office\Templates\Normal.dot
Title:	Applications of ADAMS V9
Subject:	
Author:	Marilyn Lee
Keywords:	
Comments:	
Creation Date:	06/25/98 2:43 PM
Change Number:	2
Last Saved On:	06/25/98 2:43 PM
Last Saved By:	Marilyn Lee
Total Editing Time:	1 Minute
Last Printed On:	08/06/98 3:38 PM
As of Last Complete Printing	
Number of Pages:	5
Number of Words:	1,707 (approx.)
Number of Characters:	9,732 (approx.)