Fluid-Structure Interaction Simulations with Abaqus/Explicit

Summary
Structures that contain fluid must be analyzed under a variety of load types to determine design effectiveness. In the design of containers and consumer products, typical loading scenarios considered include drop testing, temperature change, pressurization, and stacking. For dynamic loading situations, Abaqus/Explicit includes a number of advanced features that allow certain types of fluid-structure interaction, including sloshing and inertial loading effects, to be modeled accurately. Abaqus has been used extensively in the consumer products and packaging industries for these types of analyses.

Background
Fluid that partially fills a structure may undergo sloshing whenever the containment structure experiences motion. In a sufficiently dynamic event the inertial loading of the fluid on the structure becomes a critical component of the analysis. As a result of the coupling between the fluid and the structure, large deformations may be experienced by both, possibly causing rupture of a closed container or fluid loss from an open container.

While not offering full computational fluid dynamics (CFD) capabilities, Abaqus/Explicit is well suited for analyzing this type of fluid-structure interaction.

Two examples of fluid-structure interaction analyses are presented in this technology brief.

The first example is a fluid containment simulation. A container consisting of a box with a lid is partially filled with fluid, and the box is given a velocity history. The purpose of the analysis is to determine if the fluid sloshing will cause the lid to lift from the box.

The second example is a 3-ft. drop test of a partially filled consumer bottle. The purpose of the analysis is to assess whether the bottle will rupture.

In both problems the effect of the inertial fluid-structure coupling on the structural response is of primary interest.

Finite Element Analysis Approach
The fluid and structure in both problems are meshed as separate bodies, with contact definitions defined at the interfacing surfaces. The general contact algorithm, which allows for very simple definitions of the contact interactions, is employed.

Key Abaqus Features and Benefits
- Explicit dynamic solution method for efficient analysis of transient, highly nonlinear problems.
- Equation of state models for fluid constitutive behavior.
- Automatic adaptive meshing to maintain fluid mesh quality.
- Robust contact algorithm.
- Material and element failure for simulation of container rupture.

Fluid containment analysis
The purpose of this analysis is to assess whether the sloshing of fluid in a partially filled box would lift the lid when a velocity history is applied to the box. Shell elements are used to represent both the box and the lid, and solid hexahedral elements are used for the fluid. The lid and the box are significantly stiffer than the fluid, so these components are modeled as rigid bodies. Figure 1, Figure 2, and Figure 3 show the meshes used for each portion of the model.

Contact is defined between the lid and the box and between the fluid and the entire container. No attachment has been defined between the lid and the box.
Abaqus has a number of contact algorithms; the general contact capability is the easiest to use and the most comprehensive. An advantage of the general contact algorithm is its ability to include the “edge-to-edge” contact at the top of the box in the contact definition. With edge-to-edge contact, geometric feature edges, perimeter edges of shell and solid elements, and segments of beam and truss elements can be included in the contact domain. This feature allows contact interactions that cannot be detected as penetrations of nodes into faces to be enforced.

The box is constrained to remain on the horizontal plane throughout the analysis; that is, it is not allowed to rotate in space or to lift from the ground. Gravity loads are applied to the lid and to the fluid; and a sinusoidal, time-varying velocity is applied to the box. The velocity history is such that the box moves only in the horizontal plane; no vertical motion has been prescribed.

The lid simply sits on the box; therefore, an airtight seal is not assumed at this interface. Consequently, there is no gas pressure in the space above the fluid. If the effect of a gas were to be included in the analysis, the following modeling approaches could be taken:

- The gas could be modeled with solid elements (using an equation of state material model) and a contact interface between the fluid and the gas.
- A surface-based hydrostatic fluid cavity could be defined for the gas.
- A simple surface pressure load could be applied to the free surface of the fluid.

**Bottle drop analysis**

The purpose of this analysis is to determine the integrity of a fluid-filled bottle when dropped from a height of 3 ft. Shell elements are used to represent the bottle, and solid hexahedral elements are used for the fluid. A single rigid element is used to model the floor. The undeformed model is shown in Figure 4.

Contact is defined between the bottle and the fluid and between the bottle and the floor. Gravity loads are applied to both the fluid and the container. The initial conditions of the bottle are consistent with a drop of 3 ft. The bottle is positioned slightly above the point of contact, and the fluid and the container are given an initial velocity of 168 in/s. The rigid floor is fully constrained.

The elastic-plastic constitutive properties of the bottle are those of high-density polyethylene (HDPE). A failure model is included for the HDPE, based on the tensile hydrostatic pressure stress in the elements. This failure model allows elements to be deleted from the mesh once failure has been detected.

The general contact algorithm will automatically eliminate failed elements from the contact domain and update contact surfaces so that the resulting surface lies on elements that have not failed; surface erosion is a key capability in modeling the contact as the fluid sloshes out of a broken container.

The base of the bottle is thicker than the walls, making the material in this region slightly more resistant to failure. This region is shown in red in Figure 5.
The effect of any gas pressure in the bottle has been ignored. In this case the size of the enclosed gas cavity is small compared to the fluid, so the effect of the gas has been assumed negligible.

The additional features common to both models are discussed below.

**Equation of state material model**

In both examples the fluid is considered as incompressible and inviscid. An equation of state material model is typically used for such applications and is chosen here. The equation of state determines the volumetric strength of a hydrodynamic material and specifies the pressure in the material as a function of density and internal energy. With this approach the deviatoric strength of the material is considered separately and can be included if viscous behavior is needed.

**Section properties of the fluid elements**

Abaqus/Explicit offers alternative kinematic formulations for solid hexahedral elements: when appropriate for the analysis, choosing a nondefault formulation can significantly reduce computational expense. For the elements representing the fluid in the present simulations, an orthogonal formulation is chosen. This formulation provides a good balance between computational speed and accuracy.

If the objective of the analyses was to determine the shape of the fluid free surface with the highest possible accuracy, the default kinematic formulation would be appropriate. However, because the inertial coupling of the fluid and structure is of primary importance, a less computationally expensive formulation can be used.

**Automatic adaptive meshing for the fluid**

Automatic adaptive meshing in Abaqus/Explicit allows it to maintain high-quality element shapes as the fluid undergoes large deformation during sloshing. While a regular Lagrangian approach could be used to model the fluid, the elements would become very distorted after a short period of time. Adaptive meshing maintains well-shaped elements, allowing for a longer simulation time by periodically adjusting the element shapes in the fluid domain.

Initially regular, relatively coarse meshes of hexahedral elements are used for the fluid. A single adaptive mesh domain that incorporates the entire fluid region is defined.

In the bottle drop example a graded smoothing objective is used so that the initial mesh gradation of the water is preserved approximately while continuous adaptive meshing is performed. In addition, the default curvature refinement weighting is increased, causing the adaptive meshing algorithm to retain more elements in areas of high concave curvature.

**Analysis Results and Discussion**

Some representative results from the analyses are presented.

**Fluid containment analysis**

Figure 6–Figure 9 display the deformed shape of the fluid at several points of the analysis.

The deformed shape plots show the large deformations achieved by the fluid as the box moves. The automatic adaptive meshing capability in Abaqus/Explicit maintains well shaped elements in the fluid, allowing the fluid to achieve high levels of deformation.
No restraint mechanism is applied to the lid; it is simply placed on the box. Since the lid is modeled as rigid, the history of the vertical displacement of the center of the lid (Figure 10) clearly shows that the sloshing induced in the fluid will cause the lid to separate from the box.

**Bottle drop analysis**

The deformation and damage sustained by the bottle are shown in Figure 11–Figure 15.

The deformed shape plots clearly show the buckling response of the bottle on impact and the instant of rupture (the failed elements have been removed from the plots). The tensile failure material model produces an output variable that indicates whether failure has occurred for each element, and the Visualization module in Abaqus/CAE can remove the failed elements from the display. As the failure propagates, it can be seen that the tear travels down the corner of the bottle and turns along the interface between the thicker base section and the thinner bottle wall.

**Conclusions**

As demonstrated in the above analyses, Abaqus/Explicit can be used to incorporate the effects of sloshing-type fluid-structure interaction into dynamic analyses. While it is generally not possible in Abaqus/Explicit to model complex fluid flow behaviors or phenomena such as free-surface interactions and splashing, inclusion of the inertial loading caused by the fluid deformation allows for a more complete simulation capability.
Abaqus References
For additional information on the Abaqus capabilities referred to in this brief, see the following Abaqus 6.10 documentation references:

- Analysis User’s Manual
  - “Explicit dynamic analysis,” Section 6.3.3
  - “ALE Adaptive meshing: overview,” Section 12.2.1
  - “Equation of state,” Section 22.2.1

- Example Problems Manual
  - “Cask drop with foam impact limiter,” Section 2.1.12
  - “Water sloshing in a baffled tank,” Section 2.1.14

- Benchmarks Manual
  - “Water sloshing in a pitching tank,” Section 1.12.7

Visit the SIMULIA Resource Center to read more Technology Briefs

About SIMULIA
SIMULIA is the Dassault Systèmes brand that delivers a scalable portfolio of Realistic Simulation solutions including the Abaqus product suite for Unified Finite Element Analysis, multiphysics solutions for insight into challenging engineering problems, and lifecycle management solutions for managing simulation data, processes, and intellectual property. By building on established technology, respected quality, and superior customer service, SIMULIA makes realistic simulation an integral business practice that improves product performance, reduces physical prototypes, and drives innovation. Headquartered in Providence, RI, USA, with R&D centers in Providence and in Suresnes, France, SIMULIA provides sales, services, and support through a global network of over 30 regional offices and distributors. For more information, visit www.simulia.com

The 3DS logo, SIMULIA, Abaqus and the Abaqus logo are trademarks or registered trademarks of Dassault Systèmes or its subsidiaries, which include ABAQUS, Inc. Other company, product and service names may be trademarks or service marks of others.

Copyright © 2007 Dassault Systèmes