DRAFT 18 Oct 2010 (no comments from Jerry yet)

Guidelines for ALE Modeling in LS-DYNA

Jim Day, LSTC

Reviewers
Ian Do, LSTC;
Jerry Farstad??, Boeing

In modeling fluid or fluid-like behavior, a Lagrangian approach, wherein the deformation of the finite element mesh exactly follows the deformation of the material, is often not suitable owing to the very large deformation of the material. Mesh distortion can become severe, leading to a progressively smaller explicit time step and eventual instability.

In contrast, an Eulerian or ALE (Arbitrary Lagrangian Eulerian) solution method, wherein the materials flow (or advect) through the Eulerian/ALE mesh which itself is either fixed in space (Eulerian) or else moving according to some user-issued directives (ALE), is much better suited to modeling of fluid or fluid-like behavior.

In LS-DYNA, Lagrangian and Eulerian/ALE solution methods can be combined in the same model and the fluid-structure interaction (FSI) may be handled by a coupling algorithm. Thus parts that deform a moderate amount, such as structural components of metals, composites, or polymers, can be modeled with Lagrangian elements whereas fluids, such as air and water, and fluid-like parts, such as birds or ice impacting at high velocity, can be modeled with Eulerian/ALE elements. Bear in mind that at very high pressure, temperature, and/or strain rate, even structural materials (metal, concrete, soil, etc.) may behave in a fluid-like manner and thus may be more suitably modeled with Eulerian/ALE elements in such cases.

This article gives an introduction to and general guidelines for modeling with Eulerian/ALE elements. Modeling of airbags is a special and complex subtopic that is not addressed in this document.

Introduction – Lagrangian vs. Eulerian vs. ALE Formulations

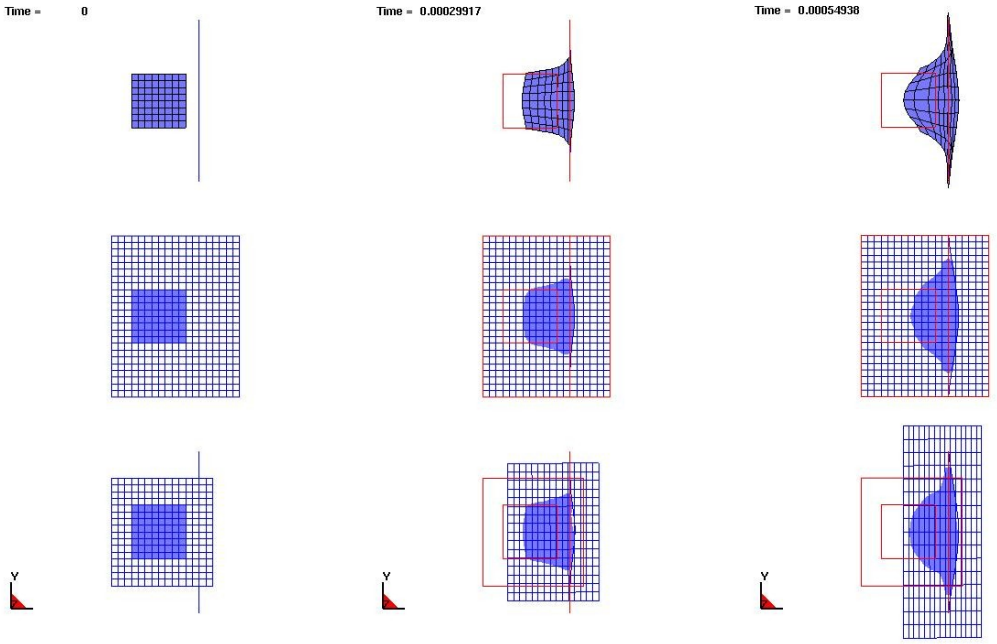
The figure below shows a 3-frame sequence of a water projectile striking a metal plate. Each row in this figure represents a different modeling approach and helps to illustrate the fundamental differences in Lagrangian (1st row), Eulerian (2nd row), and ALE formulations (3rd row). The solid blue portion represents the water projectile. The red outline is for reference only and marks the initial, undeformed location of the parts. The LS-DYNA input deck for the model that produced this figure is available at
http://ftp.lstc.com/anonymous/outgoing/jday/aero/3in1_impacting_plate.k.

LAGRANGIAN

In an all-Lagrangian approach (3-frame sequence shown in the top row), the nodes move directly along with the material and thus elements and materials translate, rotate, and deform together. Material does not cross element boundaries and thus the mass of material within each Lagrangian element never changes.

EULERIAN

For Eulerian (and ALE), the mesh is simply a logical computational space and the materials may deform or flow through that space, moving from element to element. In the sequence shown in the middle row of the figure, the metal target plate remains Lagrangian, but now the mesh of the water and surrounding void is Eulerian which means this mesh remains fixed in space. The water (shown in solid blue/dark) can move and deform within the fixed mesh. As the simulation progresses, the materials (water and void) cross element boundaries, i.e., with each time step some small amount of each material may flow or advect out of one cell (or element) and into an adjacent cell. At any given point in time, each Eulerian element may contain a mixture of water and void, hence the term "multi-material" is used in describing the element formulation. The process by which history variables, e.g., stresses, are calculated within a mixed element is beyond the scope of this modeling document.



ALE

The third approach, shown in the bottom row of the figure, employs an ALE formulation in modeling the water projectile and surrounding void. Again, the metal target is Lagrangian. Unlike the Eulerian case in which the water and void mesh remain fixed, the ALE mesh is directed to move in some prescribed manner as the solution progresses. Thus Eulerian is a special case of ALE wherein the prescribed reference mesh velocity is zero. Subsequently we may refer to this general Eulerian/ALE class of methods as simply "ALE". Unlike the wholly

Lagrangian case in which the mesh and material move exactly together, the ALE mesh and the material do not move exactly together. Thus material advection across element boundaries is still required but the amount of material advected each time step is generally less as compared to the Eulerian approach if the mesh is directed to move in a similar direction as the material. Generally, especially in coarser meshes, the less material that is advected per time step, the more accurate the simulation, so we see that in certain situations ALE may hold an advantage over Eulerian. An additional advantage of ALE is that because the mesh can be directed to approximately follow along with the fluid material(s), generally fewer elements are needed as compared to the Eulerian approach. In other words, the entire spatial domain covered over the course of the simulation need not be meshed at the outset. It should be noted however that as the ALE mesh moves, any material entering the mesh domain (e.g. air) may not have exactly the proper thermodynamic state at its flow condition, as the flow field solution is not known a priori.

For both the Eulerian and ALE methods, parts (defined by the *PART command) correspond to groups of elements and not to particular materials, except in some cases at t = 0.0. ALE multi-materials groups (AMMGs), rather than parts, track the materials in the time and space domains.

MESH SMOOTHING

There is a subclass of ALE modeling referred to as mesh smoothing in which the mesh conforms to the exterior boundary of the ALE material and the elements are reshaped using any of several smoothing algorithms. After the elements are smoothed, material advection occurs. Although this smoothing approach is available in LS-DYNA, it is less general and less robust than the case in which the ALE mesh need not conform to the material boundaries. Thus the ALE smoothing approach in LS-DYNA is not discussed any further in these modeling guidelines. Instead, when ALE is discussed, focus will be on the general ALE approach.

FSI

When Euler or ALE parts are required to interact with Lagrangian parts, some form of coupling (or fluid-structure interaction, FSI) feature must be defined. (The exception is if nodes are shared between the ALE mesh and the Lagrangian mesh at their juncture – a practice which is generally not recommended.) There are three approaches to ALE-to-Lagrangian coupling. The first two approaches fall under the heading of *constrained_lagrange_in_solid and are constraint-based and penalty-based, respectively. The third approach, *ale_fsi_projection, is a relatively new, constraint-based method which does not conserve energy but which has advantages in certain situations. Of the three methods, penalty-based coupling is generally favored as it conserves energy and is robust, although it is subject to leakage. The coupling commands in LS-DYNA are discussed in more detail later in this article.

LIMITATIONS

There are some limitations to the ALE approach to consider.

The ALE compressible flow solver in LS-DYNA has been developed with the intent of simulating short duration problems with high pressure and velocity gradients. The solver is not well suited to problems driven by low pressure gradients such as in acoustic problems nor is it suited to long duration problems lasting more than a few seconds. The time duration limitation

is driven by the very small time step size characteristic of explicit time integration. This time step is roughly the smallest time it takes a stress wave to traverse any element (Courant stability criterion). In the case of ALE, time step size may be further limited by the penalty stiffness of the ALE-Lagrange coupling. Tuning of this coupling stiffness is often necessary to achieve proper FSI. If the coupling is too soft, leakage may be excessive. If the coupling is too stiff, the time step will drop and the computation time will go up accordingly.

The ALE solver is based on conservation laws with the material behavior uncoupled from the system of governing equations. This is the main deviation from the traditional CFD-type Navier-Stokes solvers where the material behavior variables may be explicitly integrated into the equations. The ALE solver does not account for boundary layer effects such as drag, etc., and it cannot handle turbulence-dominated processes. Effects of fluid viscosity derive solely via the material model, e.g., via the dynamic viscosity MU in *mat_null.

ALE is relatively expensive as compared to Lagrangian owing to the additional advection, interface reconstruction, and coupling computations.

Advection associated with the ALE solver is inherently dissipative to some extent, e.g., pressure amplitude emanating from detonation of explosive tends to be underpredicted, especially when first order accurate advection is employed (METH = 1 in *control_ale). Nonphysical energy dissipation is generally reduced when second order accurate advection is employed (METH = 2) but there is some additional computational cost. Refining the mesh will also help to reduce energy loss but again there is additional computational cost. It is typically beneficial to go through the debugging process using METH=1 and perhaps testing METH=2 in the last stage, unless the analysis requires higher order advection.

Results from the ALE solver may exhibit some slight to moderate mesh biasing effects. For example, a pressure wave originating from a point source in a fluid may become less and less spherical as the distance from the point source increases. This mesh biasing effect is reduced or eliminated when the mesh lines run parallel to and perpendicular to the primary direction of wave propagation. For example, in simulating detonation from a point source, a spherical mesh with a small “butterfly” core region centered on the detonation point can be used to good effect.

ALE Element Formulation

When two or more fluids or fluid-like materials (empty space counts as one material) are to be modeled using the ALE approach in LS-DYNA, the recommended element formulation for those materials is the multi-material ALE formulation (ELFORM = 11 in *section_solid). Although there are other ALE element formulations (ELFORM 5, 6, and 12), those are of interest perhaps only in an academic sense and will not be discussed here.

To review, as the ALE materials flow through the ALE mesh, the material boundaries or interfaces in general do not coincide with the mesh lines. These material interfaces are internally reconstructed each time step based on the volume fractions of the materials within the elements. Each material which the user wants to track individually must be assigned a unique ALE multi-

material group (AMMG) ID via the command *ale_multi-material_group. Parts sharing the same material properties may be included in the same AMMG ID or, at the user's discretion, can be distributed into separate AMMG IDs to allow for independent tracking of each group. Materials which do not share the same material properties cannot be part of the same AMMG.

Generally some portion of the ALE mesh is initially devoid of material or else is initially filled with a gas at STP condition (standard temperature and pressure). This void or pseudo-void provides space into which other, higher density materials may be transported as the simulation progresses. In our earlier example, water moved with time into elements initially devoid of material. Space initially devoid of material (and thus having zero mass, zero pressure, etc.) is modeled with *mat_vacuum. If the space is occupied by air or some other ideal gas with nonzero density, with or without nonzero pressure, a material model and an equation-of-state appropriate for such a gas, e.g., *mat_null and *eos_ideal_gas, should be assigned to that space.

Motion of the ALE mesh is controlled by the family of command(s) *ale_reference_system_*option*. Without such a command, the ALE mesh will remain stationary thus becoming the special case of Eulerian. Using these commands, one can prescribe the motion of the ALE mesh in a very specific and/or predetermined manner, or the mesh motion can be made to approximately follow the mass-weighted average velocity of the ALE materials. The latter option is perhaps the most common and useful choice and is invoked by setting PRTYPE=4 in *ale_reference_system_group.

Since the ALE method allows for materials to flow between elements and the user has direct control over the ALE mesh motion, ALE element distortion is generally of no concern. It follows that hourglass deformation is less of an issue in the case of ALE than in the case of Lagrangian, and the need for hourglass forces to restrict hourglass deformation is much reduced or eliminated. For materials modeled as ALE, hourglass formulation 1 and a much reduced hourglass coefficient, e.g., 1.0E-6 or less, are recommended to prevent application of inappropriate hourglass forces. This recommendation is especially true in the case of modeling gases and liquids. Starting in version 971 R3.1, the default hourglass coefficient for all parts with ELFORM=11 is set to 1.E-06. The default hourglass control can always be overridden by the user using *hourglass and HGID in *part. Such an override may be appropriate in the case of solid (non-fluid) ALE materials.

Meshing

Hexahedral elements with reasonable aspect ratios should be used for the initial ALE mesh. Degenerate element shapes such as tetrahedrons and pentahedrons should be avoided as they may lead to reduced accuracy at best and perhaps numerical instability during the advection. Bear in mind that use of *ale_reference_system may affect the element shapes as the solution progresses. If element shapes become unreasonable, controls in the *ale_reference_system_*option* command(s) may need to be adjusted to maintain reasonable element shapes.

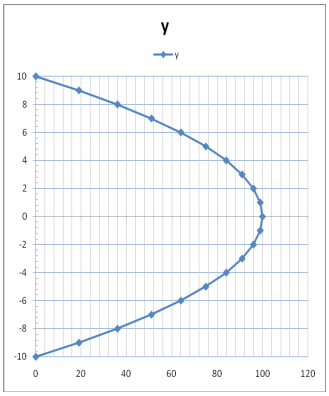
An initial ALE mesh may be constructed using one of the following two approaches:

- The initial mesh of the ALE domain may be constructed to conform to the materials, i.e., there are no mixed (or partially filled) cells in the initial configuration. Mesh lines follow the outer contour of each AMMG.

- A regular, orthogonal mesh of the ALE domain may constructed with no restriction that mesh lines follow the outer contour of each AMMG. In this case, there will likely be elements containing more than one AMMG. For these mixed elements, the initial volume fractions of AMMGs must be prescribed via *initial_volume_fraction(_geometry). This command has a "geometry" option that automates the assignment of initial volume fractions to ALE elements. At the conclusion of the automatic assignment of initial volume fractions, LS-DYNA writes a file containing the *initial_volume_fraction data for each ALE element before continuing with the simulation. This file can be utilized in subsequent runs in lieu of the *initial_volume_fraction_geometry command, thereby speeding up initialization.

What constitutes an appropriate degree of refinement for the ALE mesh is at least partially dictated by the geometric characteristics of the Lagrangian parts. Though not a requirement, a reasonable goal is to have the ALE elements be nearly the same size as the Lagrangian elements where coupling is to take place.

If ALE material is to flow through any passages in the Lagrangian mesh, use at least 5 to 10 elements across the passage width in order to adequately resolve the flow. Consider as a guideline using a number of elements across the passage equal to the points necessary to resolve a parabolic shape such that the area of the parabola is preserved to the user's required accuracy.



As stated under the limitations section above, results may exhibit some mesh bias. If these effects appears to be significant, reconstruction of the initial mesh and controls on mesh movement (*ale_reference_system_*option*) may be warranted.

Coupling Lagrangian Surfaces to ALE Materials

Most often, in Fluid-Structure Interaction (FSI) problems modeled with LS-DYNA, the fluids (and sometimes other materials that behave in a fluid-like manner) are modeled with ALE hexahedrons and the structure is modeled with Lagrangian shells or solids. In such a model, the Lagrangian mesh usually does not share nodes with the ALE mesh. Rather, the two meshes interact via a coupling algorithm defined with the command *constrained_lagrange_in_solid. This coupling serves to generate forces that resist penetration of the ALE material through the Lagrangian parts. Coupling is a key and sometimes complex aspect of ALE modeling. Some recommendations for using *constrained_lagrange_in_solid for coupling are provided below.

Let us consider some of the more critical parameters of the *constrained_lagrange_in_solid card.

```
$---------------------------------------------------------------------------
*CONSTRAINED_LAGRANGE_IN_SOLID
$    slave    master     sstyp     mstyp     nquad     ctype     direc     mcoup
         1       200         1         0         2         4         2        -1
$    start       end      pfac      fric    frcmin      norm   normtyp  DAMPFRAC
       0.0       0.0  0.100000       0.0  0.300000
$       cq      hmin      hmax     ileak     pleak   lcidpor
       0.0       0.0       0.0         0   0.10000
$4A IBOXID   IPENCHK   INTFORC  IALESOFT    LAGMUL    PFACMM      THKF
         0         0         1         0         0         0       0.0
$---------------------------------------------------------------------------
```

The slave side parameters SLAVE and SSTYP identify the Lagrangian part(s) or segment sets to be considered in the coupling. The master side parameters MASTER and MSTYP identify, by part or part set ID, the ALE mesh that will interact with the slave side. Again, the master side identifies mesh but not material. Together, SLAVE, SSTYP, MASTER, MSTYP define the overlapping computation domains (Lagrangian and ALE) that the code will search for interaction. This does not yet specify which ALE material(s) flowing through the ALE domain are to be coupled to the Lagrangian structure.

A separate parameter MCOUP identifies the specific ALE materials, or more precisely, the AMMGs that will interact with the slave side.

To summarize coupling thus far, for coupling forces to be developed on a Lagrangian surface, that surface must (1) reside on the slave side of a *constrained_lagrange_in_solid, (2) that surface must be spatially overlapping a portion of the ALE mesh identified by the master side, and (3) that surface must be penetrating at least one of the AMMGs identified by the parameter MCOUP. See below for more discussion of MCOUP.

The parameter NQUAD determines the number of coupling points distributed over each Lagrangian slave segment. If NQUAD=2 (default), then there are 2x2 = 4 coupling points on each Lagrangian slave segment. The coupling algorithm looks for penetration of any ALE material meeting the conditions of MASTER, MSTYP, and MCOUP across each of the coupling points. If penetration at a coupling point is found, coupling forces are applied to counteract

penetration. The larger the value of NQUAD, the more expensive the coupling and the more likely the coupling forces will be excessive. If the Lagrangian slave segments are approximately the same size as or smaller than the Eulerian/ALE element faces, NQUAD=2 will generally suffice. If the Lagrangian slave segments are coarser/larger than the ALE element faces, NQUAD may need to be raised to 3 or higher to provide proper coupling.

The parameter CTYPE identifies the coupling algorithm employed. In most applications, penalty-based coupling is more robust and is therefore preferred over constraint-based coupling. Thus CTYPE should generally be set to 4, or in the case where the Lagrangian slave side is comprised of solids which may be eroded due to material failure criteria, CTYPE should be set to 5. There are other CTYPEs that allow for physical porosity of the Lagrangian surfaces, e.g., as in the case of an airbag or parachute, but a discussion of modeling porosity effects is outside the scope of this document. For the special case of coupling Lagrangian beam elements within a Lagrangian solid mesh, e.g., as used in coupling rebar to concrete, the constraint-based coupling algorithm should be used (CTYPE=2).

The parameter DIREC should generally be set to 2 as this most often best represents the physical nature of the interaction. Furthermore, it is also the most reliable and robust option. With DIREC=2, normal direction coupling occurs only in compression. Tangential coupling, associated with friction between materials, is controlled separately via the parameter FRIC.

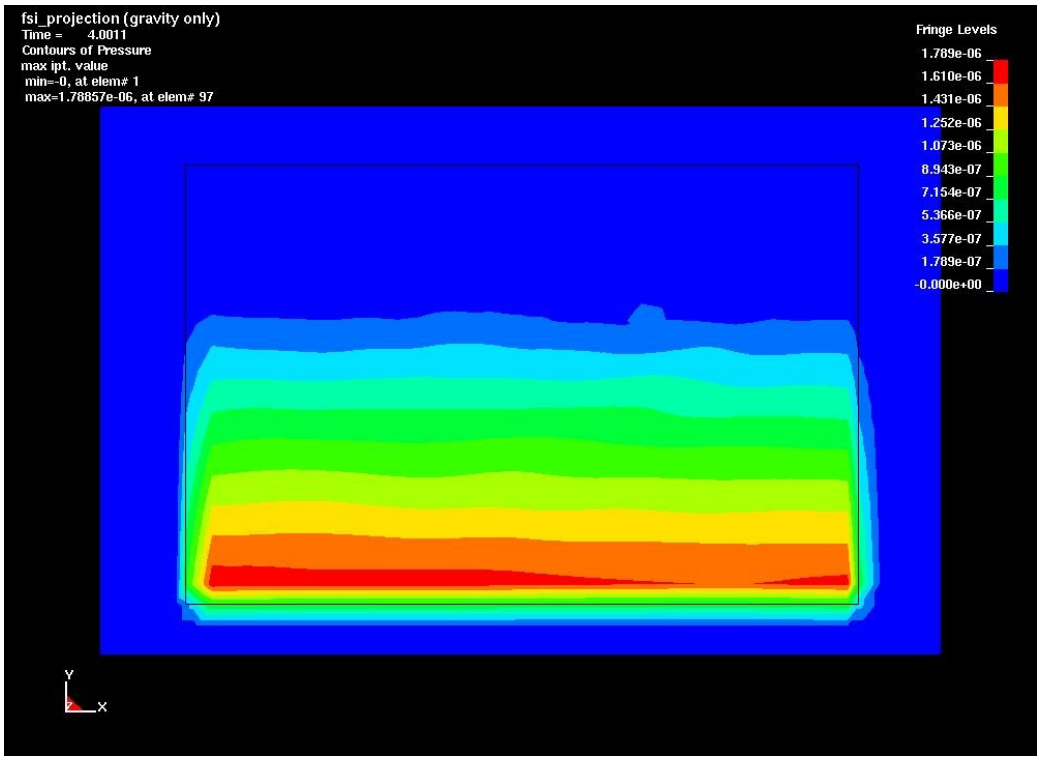
The parameter MCOUP defines the AMMG(s) to which the Lagrangian slave side is coupled. In cases where one AMMG dominates the forces imparted to the Lagrangian structure and the forces from any other AMMGs can be neglected, MCOUP should be set to 1. This might be the case where the density of one AMMG is far greater than the density of the other AMMGs. In cases where the effects of two or more AMMGs need to be considered in the coupling, MCOUP can be set to a negative number. In this case, |MCOUP| identifies a set of one or more AMMGs to be considered in the coupling. That set is defined using the command *set_multi-material_group_list.

When the slave side of the coupling is comprised of Lagrangian shells or of a segment set comprised of Lagrangian element faces, an additional requirement of successful coupling is that the slave shell/segment normals must point *toward* the AMMGs to which coupling is desired. If the slave side normals happen to point *away* from the AMMGs, these normals can be automatically reversed and the situation remedied by setting the parameter NORM=1. Note that setting NORM to 1 reverses *all* the normals of the Lagrangian slave segments. Thus it is imperative that the slave segment normals are at least consistently oriented either pointing toward (NORM=0) or away from (NORM=1) the ALE material.

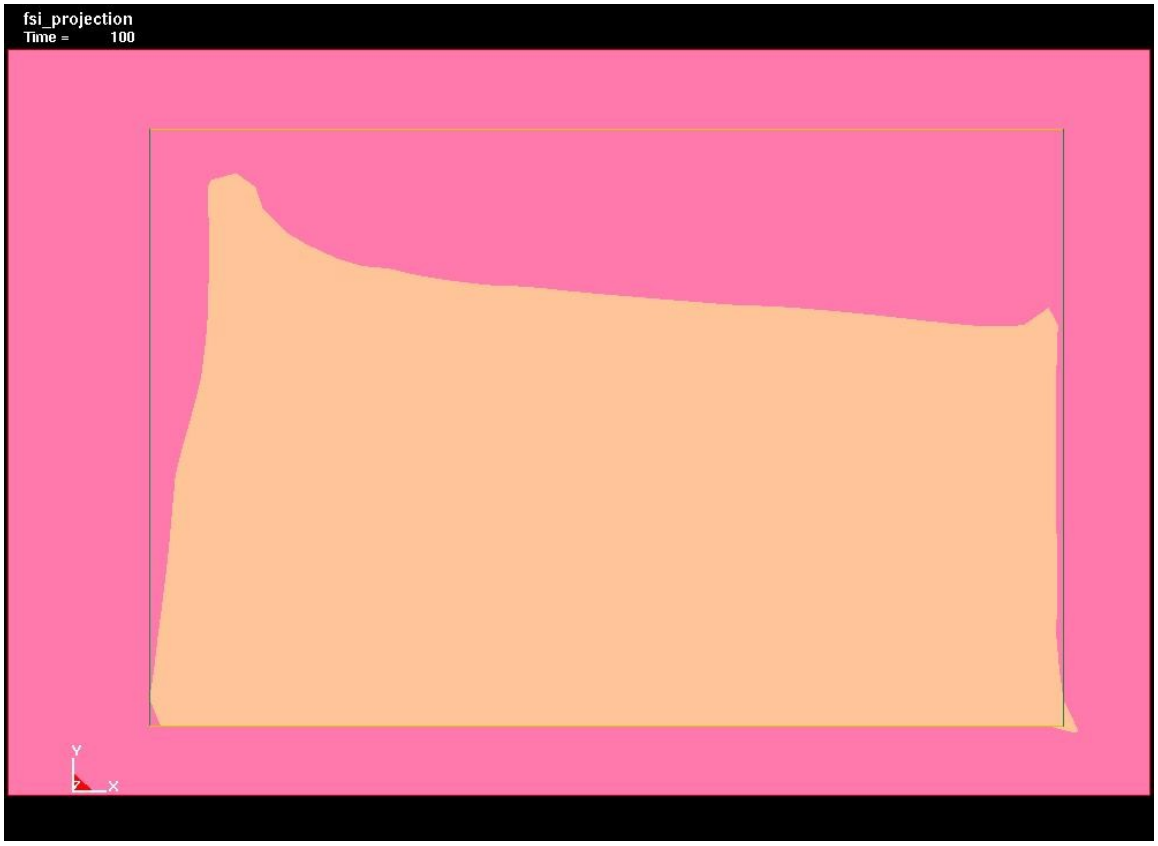
Leakage is an undesireable condition whereby coupling does not prevent unreasonable penetration of ALE material through Lagrangian surfaces. Problems of leakage can be identified visually when postprocessing as described in a later section. A small amount of leakage is to be expected for penalty-based coupling and can be tolerated, just as in the case of small penetrations seen for penalty-based contact. The following modifications to the coupling input are presented as possible remedies to excessive leakage.

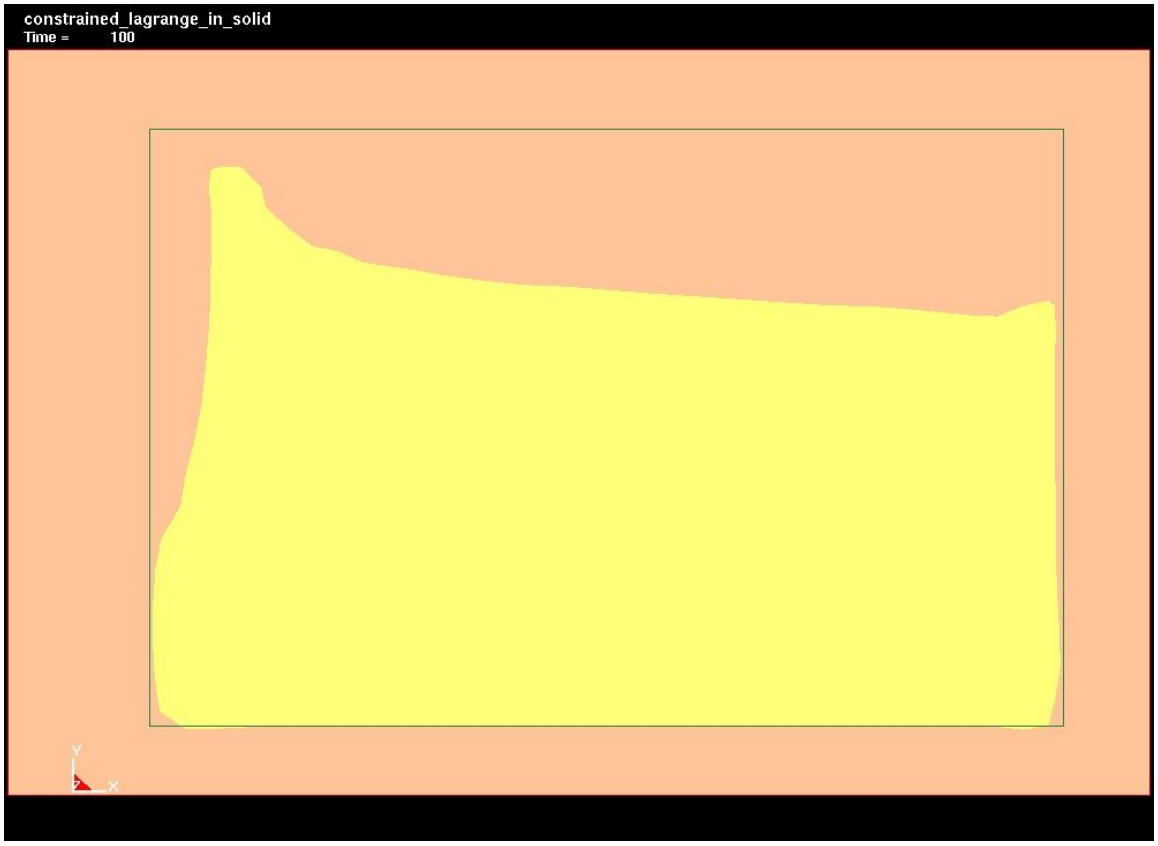
- Increase the value of NQUAD if it is suspected that there are too few coupling points on the Lagrangian segments. Be judicious here because increasing NQUAD drives up the cpu time.
- When coupling to a shell surface, assign one AMMG ID to the ALE material on one side of the shell surface and a different AMMG ID to the ALE material on the opposite side. Of course, this practice is a requirement if there are two physically different materials to either side. The point is that this guideline applies even when the same physical fluid exists on both sides of the shell surface.
- Use a separate *constrained_lagrange_in_solid command for each AMMG. This will require the use of a negative MCOUP value and a *set_multi-material_group_list command for each *constrained_lagrange_in_solid command.
- An appropriate coupling stiffness is key to good coupling behavior when CTYPE=4 or 5. In most cases, the default penalty stiffness (PFAC=0.1) works fine and this should be your starting point. If it becomes clear that the default coupling stiffness is inadequate, simply increasing PFAC (by 5 or 10 times) might resolve the problem. A more logical approach is to set PFAC to a negative integer which tells LS-DYNA that the coupling stiffness comes from curve |PFAC| wherein the abscissa is penetration distance and the ordinate is coupling pressure. *Define_curve should be used to define curve |PFAC|. (Let's say PFAC=-20. Then curve 20 defines coupling pressure vs. penetration distance.) A rule-of-thumb in defining the curve is to define two points: (0,0) and ($1/10^{th}$ the ALE element dimension, maximum pressure observed in the ALE mesh near the leakage site). Be aware that an increase in coupling stiffness may result in a smaller time step size. Just as far too small a coupling stiffness has detrimental effects, so does far too great a coupling stiffness.
- For coupling of ALE gases to Lagrangian parts (low-density-to-high-density materials), it may help to set the parameters ILEAK=2 and PFACMM=3.

As a final word in modeling coupling between ALE and Lagrangian parts, there is a coupling method that may serve as a preferred alternative to constrained_lagrange_in_solid in some cases. *Ale_fsi_projection uses a constraint-based approach, projecting the nearest ALE nodes onto the Lagrangian surface. Coupling can be in all directions, in tension and compression only, or in compression only. Energy is not conserved in this approach but it has been shown to be effective in coupling fluid to tank walls in a sloshing tank simulation. An example in which only gravity is applied to develop hydrostatic pressure in a tank of water is provided in http://ftp.lstc.com/anonymous/outgoing/jday/aero/init_water_coupled_to_tank_fsi_projection.k . The figure below shows the hydrostatic state at the end of the simulation.



In the next two examples, the container moves horizontally to introduce sloshing of the water. http://ftp.lstc.com/anonymous/outgoing/jday/aero/init_water_coupled_to_tank_fsi_projection_with_sloshing_2couplings.k uses *ale_fsi_projection to couple the water to the tank. http://ftp.lstc.com/anonymous/outgoing/jday/aero/init_water_coupled_to_tank_ with_sloshing.k uses penalty-based *constrained_lagrange_in_solid to couple the water to the tank. The two figures below show similar results from the two simulations.


fsi_projection
Time = 100
Y
X


constrained_lagrange_in_solid
Time = 100
Y
X

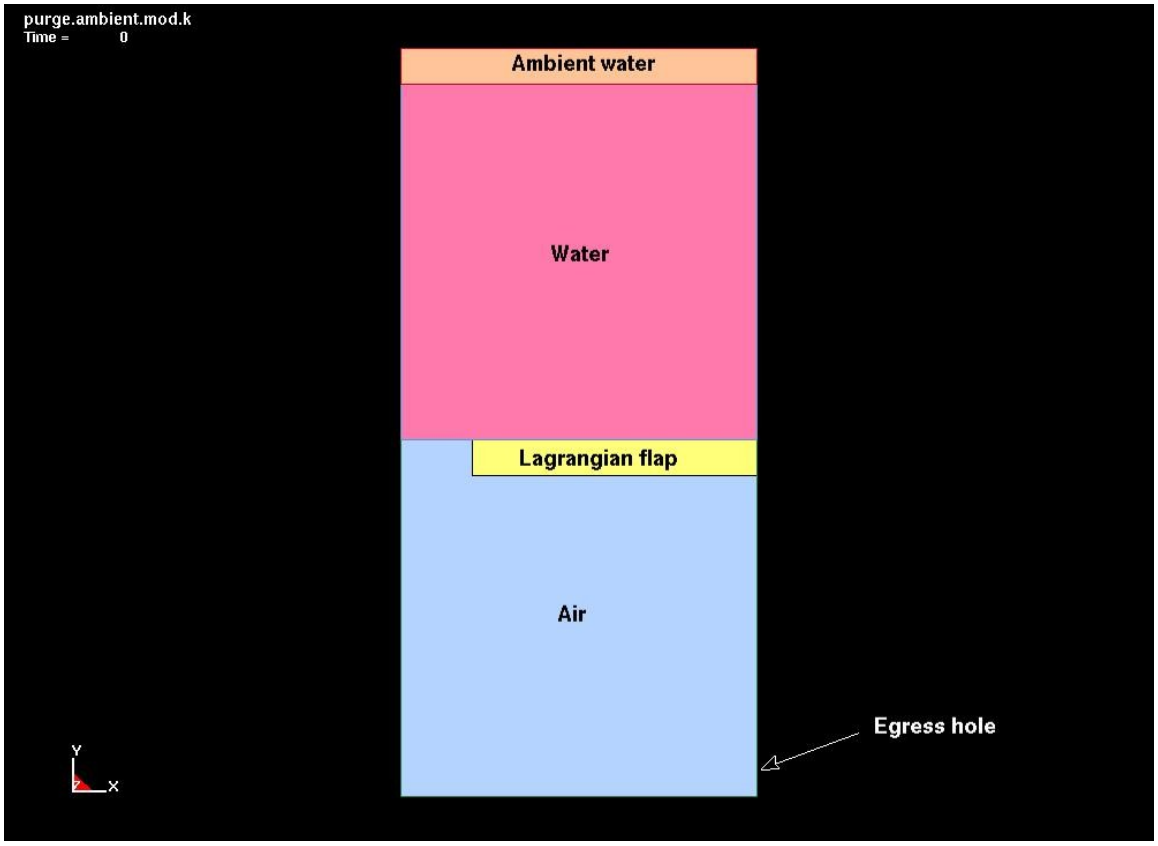
Modeling Inflow and Outflow Conditions

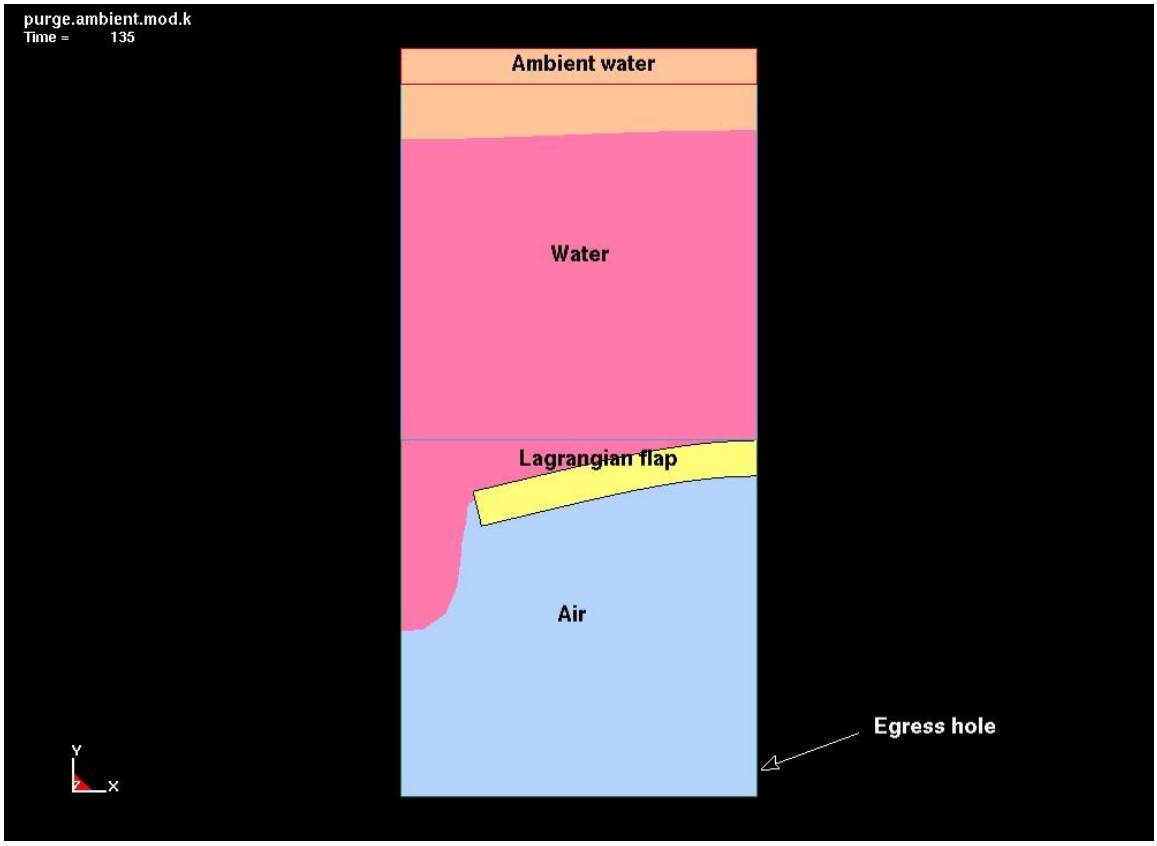
In addition to setting ELFORM to 11, setting AET to 4 in *section_solid invokes a reservoir (or ambient) type element option in the ALE formulation. The user may dictate pressure to such elements by prescribing the thermodynamic condition of the element, either as unvarying with time by simply defining E0 and V0 in the *eos (equation-of-state) input or as a function of time via *boundary_ambient_eos. Thus to model a prescribed inflow or prescribed outflow of material, one or two layers of ALE elements on the exterior of the mesh at the inflow (outflow) region is assigned a unique PART ID so that AET may be set to 4 for that layer. If the inflow or outflow conditions include a known flow velocity into or out of the ALE mesh, that velocity is prescribed by applying *boundary_prescribed_motion_node to the exterior nodes at the inflow/outflow region.

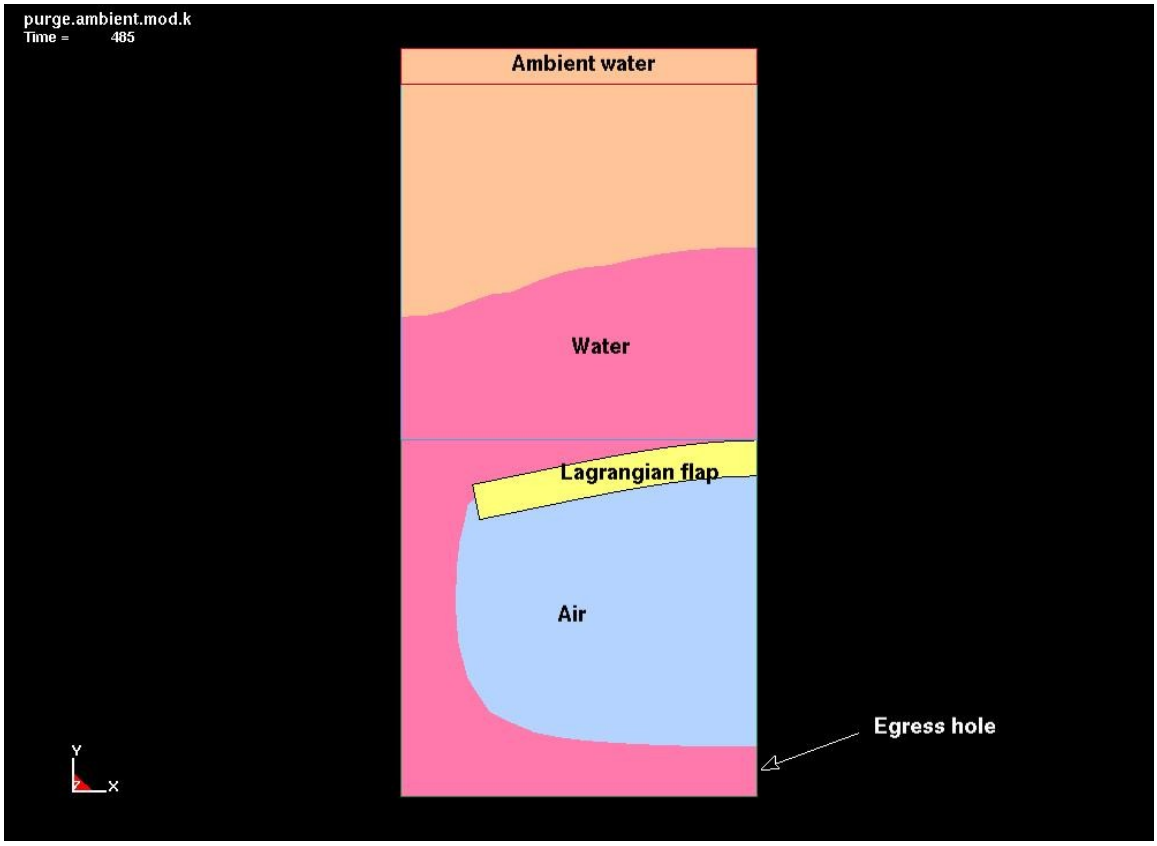
To model unprescribed (unknown) outflow, AET may be left as 0 (default) in which case outflow is calculated by LS-DYNA.

Do not attempt to assign values other than 0 or 4 to the parameter AET.

An example illustrating prescribed inflow is found at http://ftp.lstc.com/anonymous/outgoing/jday/aero/purge.ambient.mod.k . The following three figures show snapshots of the simulation. In this example, inflow of water into an empty container is diverted by a rubber flap modeled with Lagrangian solids. The rigid “container” is simulated via nodal constraints, i.e., the container is not represented by elements. An egress hole in the container is included by leaving some of the exterior nodes in the lower righthand corner unconstrained in the horizontal direction. By virtue of their ambient inflow designation (AET=4), the pressure is prescribed in the top layer of elements and that, together with gravity loading, serves to drive the simulation. Because *boundary_ambient_eos is not used in this example, the prescribed pressure in the ambient elements is a constant value, determined from the initial condition parameters in the equation-of-state.


purge.ambient.mod.k
Time = 0
Ambient water
Water
Lagrangian flap
Air
Egress hole
Y
Z
X


purge.ambient.mod.k
Time = 135
Ambient water
Water
Lagrangian flap
Air
Egress hole
Y
Z
X

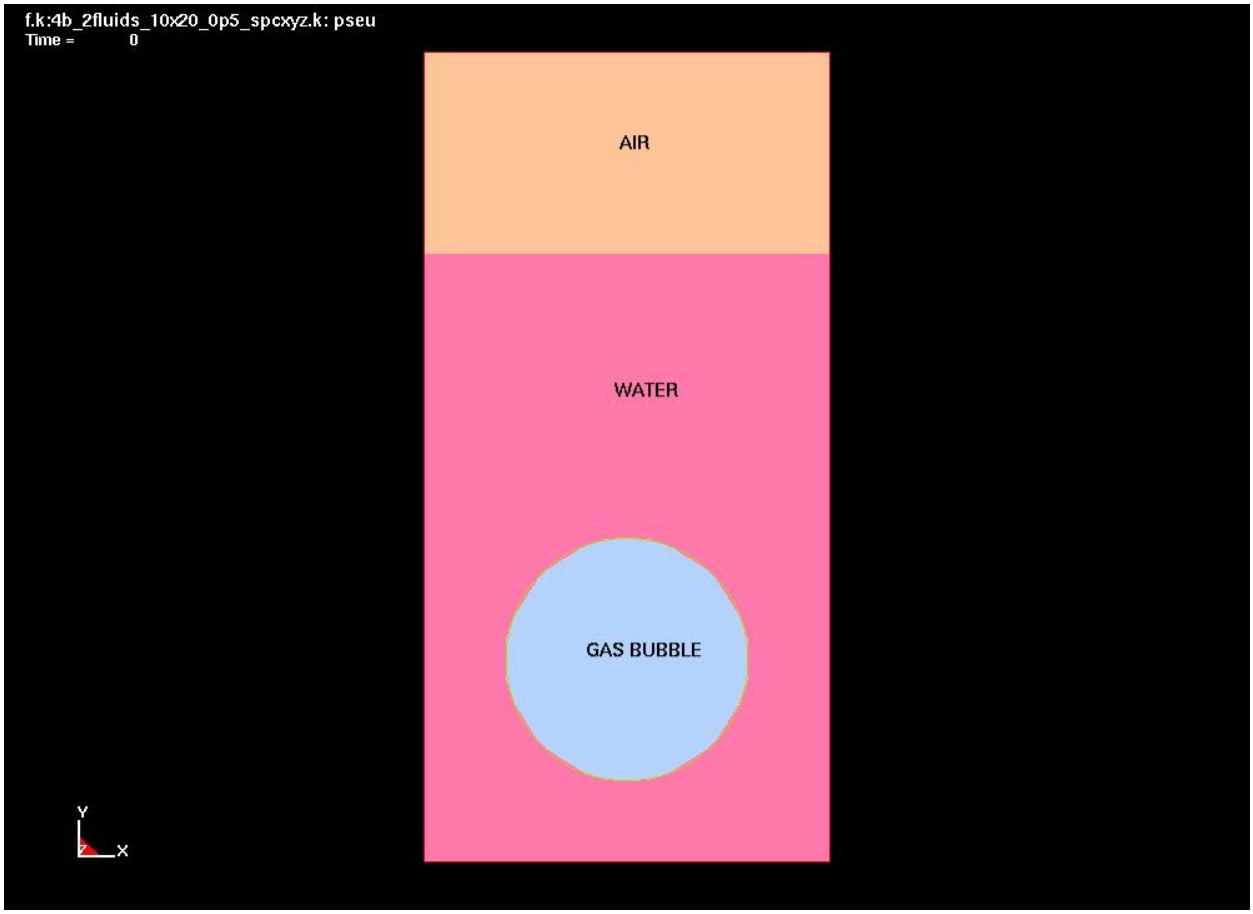

purge.ambient.mod.k
Time = 485
Ambient water
Water
Lagrangian flap
Air
Egress hole
Y
Z
X

Initializing Pressure in ALE Materials

In many situations, an initial pressure field in one or more of the ALE materials in known, e.g., atmospheric pressure in air or hydrostatic pressure in water. If the pressure field is uniform as in air at atmospheric pressure, EO and V0 in the *eos input is sufficient to initialize the pressure. In such a case, as mentioned earlier, exterior segments must also have an applied pressure to equilibrate the internal pressure, either via *load_segment or via PREF in *control_ale.

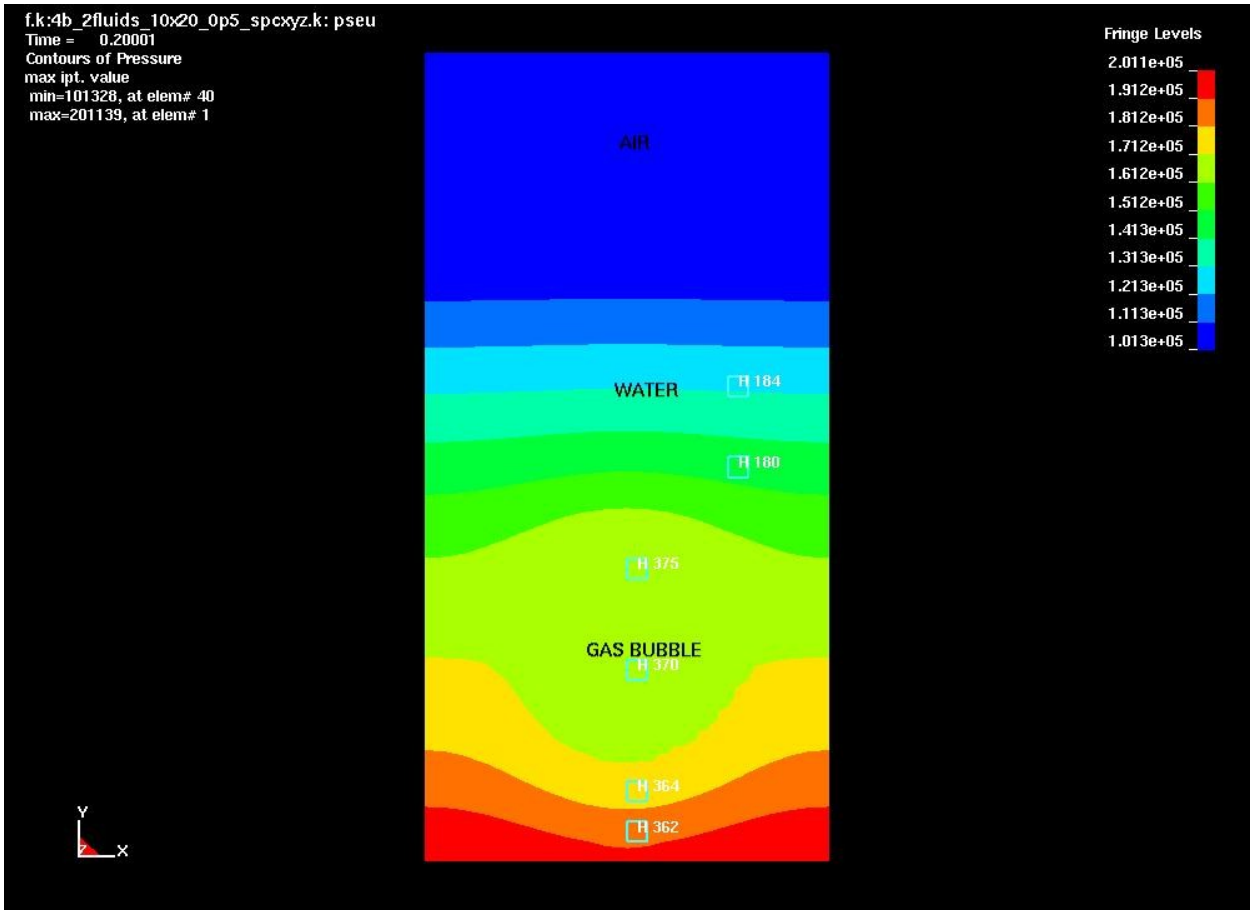
In the more complex case where pressure varies with depth as in the case of water, the command *initial_hydrostatic_ale can be used as an aid to greatly reduce the time it takes to initialize the hydrostatic pressure and reach a steady state condition in the fluid.

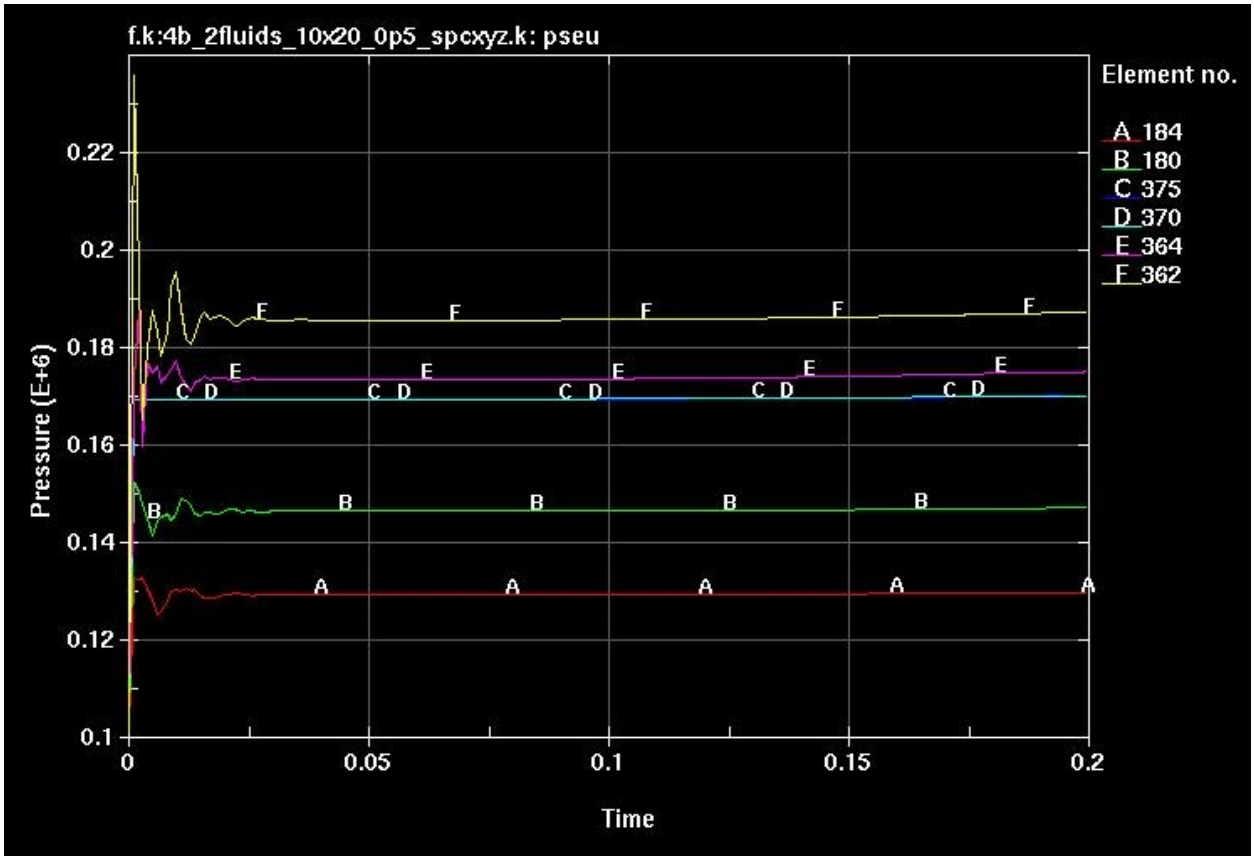
http://ftp.lstc.com/anonymous/outgoing/jday/aero/f_damp300_bub.k is an example of a pool-like condition without inflow or outflow conditions. Here, a gas bubble initially resides below the surface of the water as shown in the figure below.



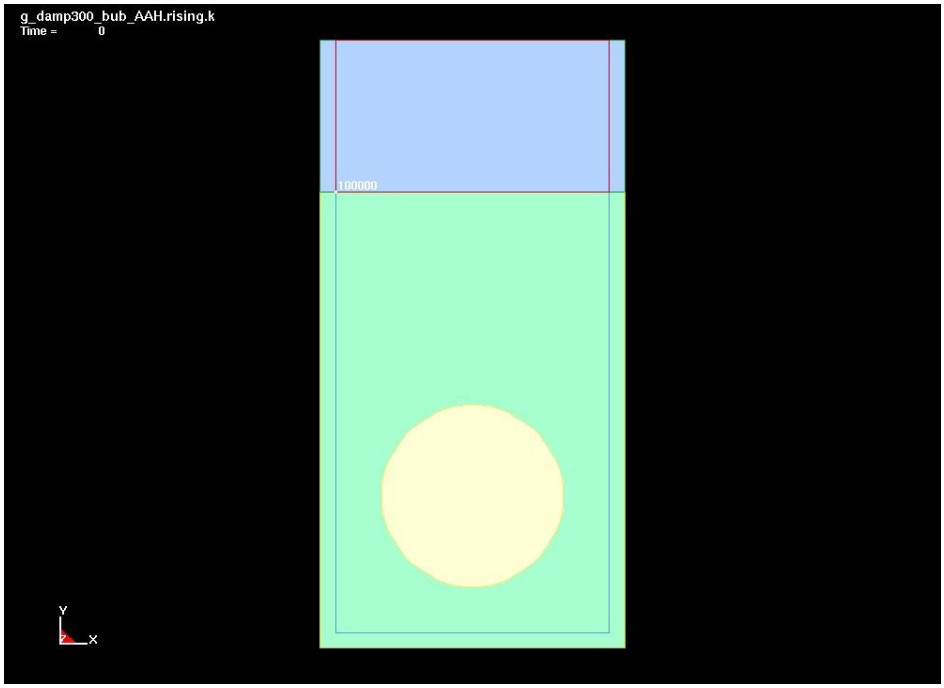
*Initial_hydrostatic_ale, in conjunction with *load_body, which applies gravity loading, and *boundary_spc, which applies the normal direction constraints to the pool walls and pool bottom, serves to quickly initialize the hydrostatic state of the fluids. In addition, this example

employs mass damping (*damping_part_mass) to remove the oscillations in pressure time histories that are otherwise seen when no damping is employed. The damping is specified as a function of time and is set to zero after achieving a steady state condition (t = 0.08 in this example) soas not to inhibit physical motion thereafter. If the termination time in the example is extended from 0.2 to 2.0, such motion is clearly evident in the form of the gas bubble rising and changing shape. Note that when mass damping is used, the value should be derived from the period of oscillation T, recognizing that critical damping is equal to 4*pi/T. Figures showing the early time results of the example are provided below. For more details of the example and of *initial_hydrostatic_ale command syntax, see pp. 14-25 of http://ftp.lstc.com/anonymous/outgoing/jday/aero/21_hydro_p_initialization_34p.pdf.

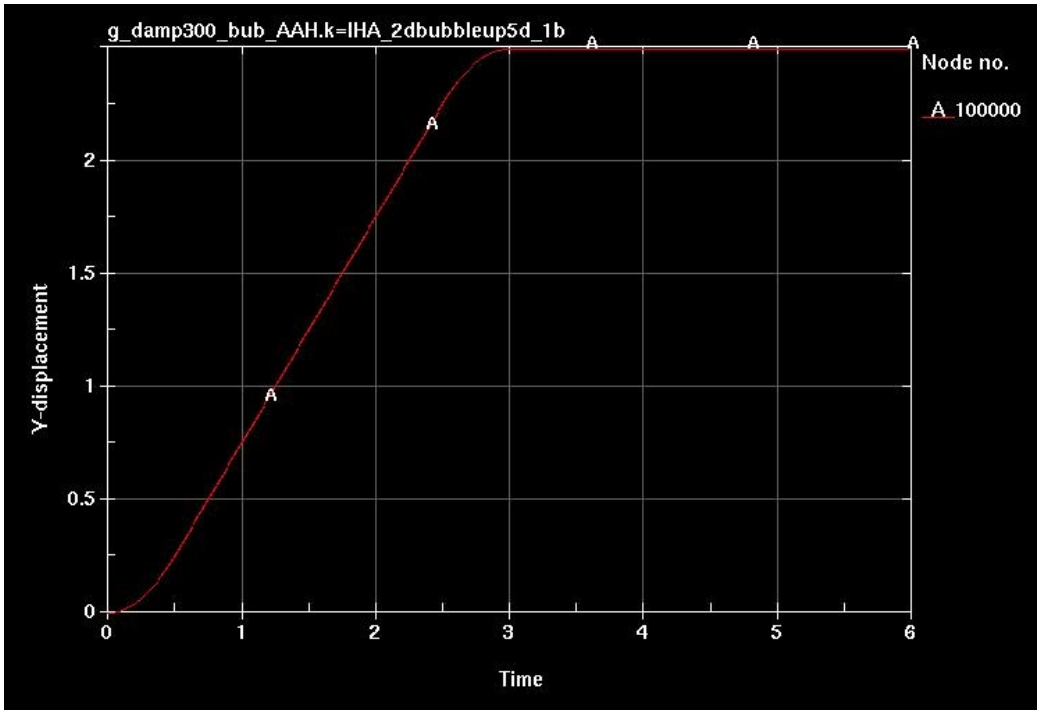




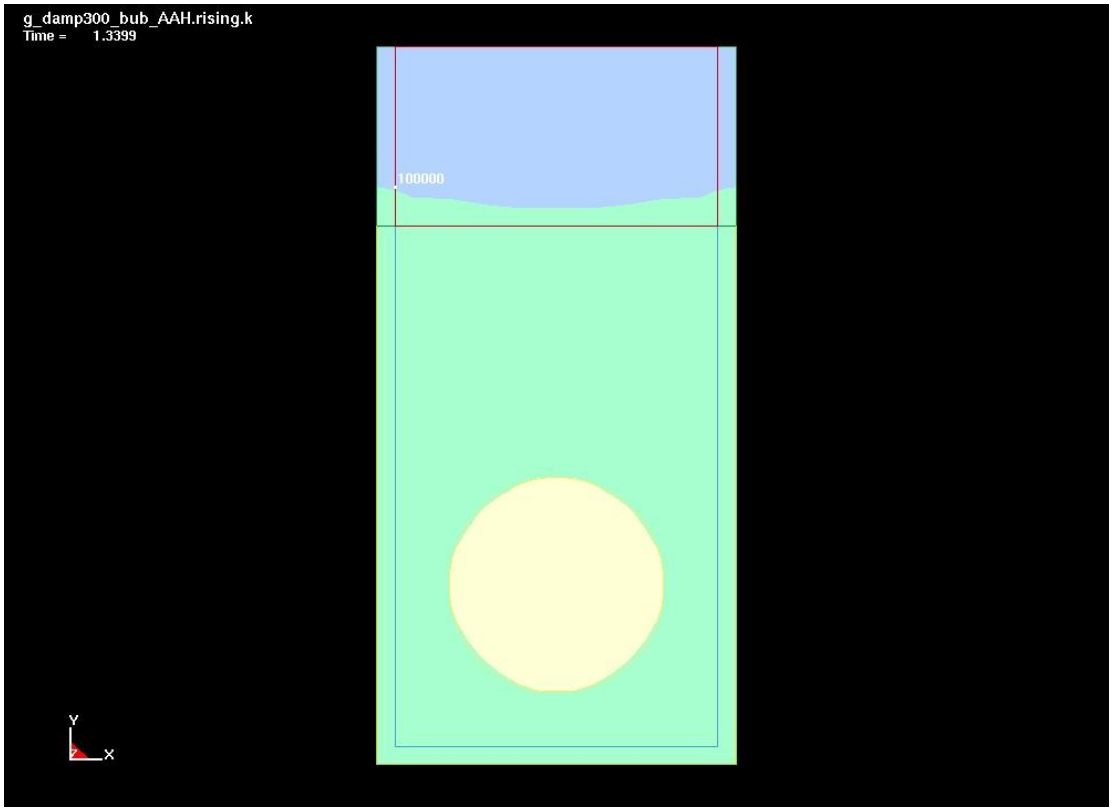
http://ftp.lstc.com/anonymous/outgoing/jday/aero/g_damp300_bub_AAH_.rising.k is an example where, instead of nodal constraints be applied to invoke a pool-like condition without inflow or outflow, a layer of reservoir/ambient elements (AET=4) are defined along the sides and bottom of the ALE mesh to permit inflow and outflow of fluid. Such a condition would be appropriate if modeling a body of water in which the water level is changing with time in a known manner in the far field, i.e., at a location where pressure is virtually unaffected by dynamic behavior, including fluid/structure interaction, taking place in the near field. In the example, a gas bubble initially resides below the surface of the water but this time is coupled to a fixed cylindrical container so that the bubble cannot rise. As before, gravity is applied via *load_body and temporary mass damping is applied via *damping_part_mass. The initial condition is seen in the figure below.

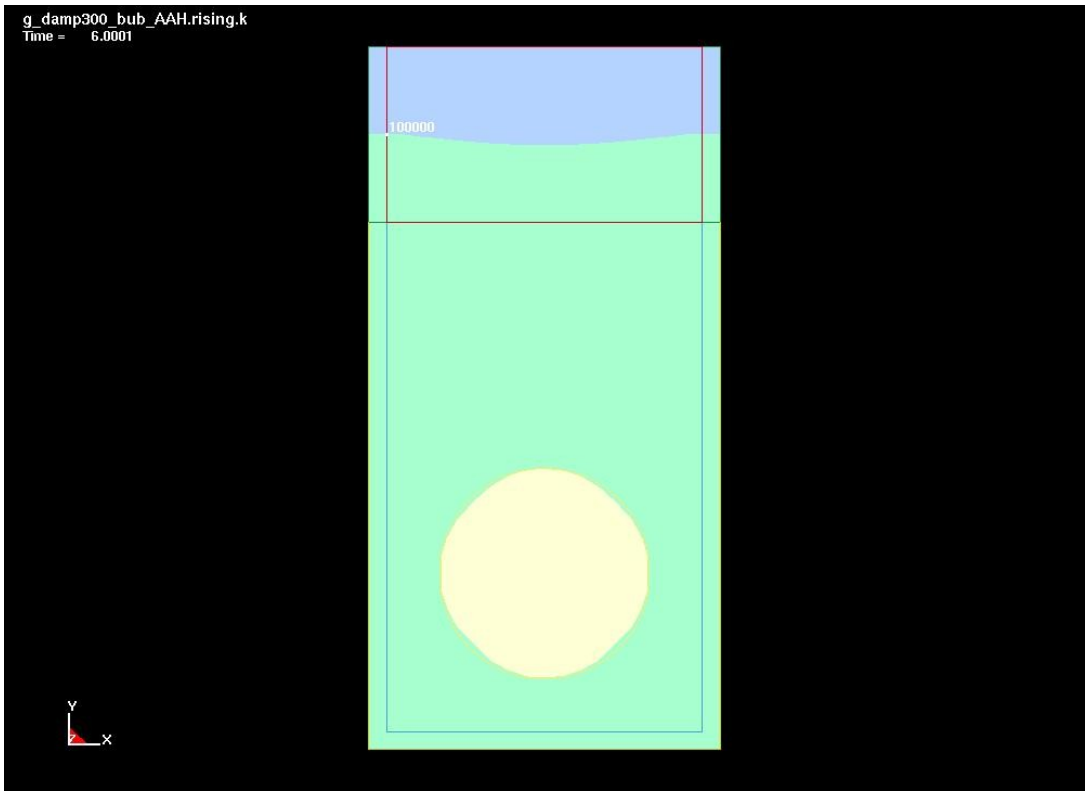


*Initial_hydrostatic_ale is again used to initialize the hydrostatic pressure in the nonambient fluid field and an additional command, *ale_ambient_hydrostatic, is used to prescribe the fluid pressure in the ambient elements along the sides and bottom of the fluid field. The command syntax of *ale_ambient_hydrostatic is identical to that of *initial_hydrostatic_ale and is shown on p. 30 of http://ftp.lstc.com/anonymous/outgoing/jday/aero/21_hydro_p_initialization_34p.pdf. The pressure in the ambient elements is automatically computed based on the location of the top surface of each fluid as tracked by nodes selected by the user. In this example, a massless node ID 100000 is given a prescribed velocity in the vertical (y) direction to control the top surface of the water in the ambient elements along the sides of the mesh. The resulting y-displacement of node 100000 is shown in the following figure.

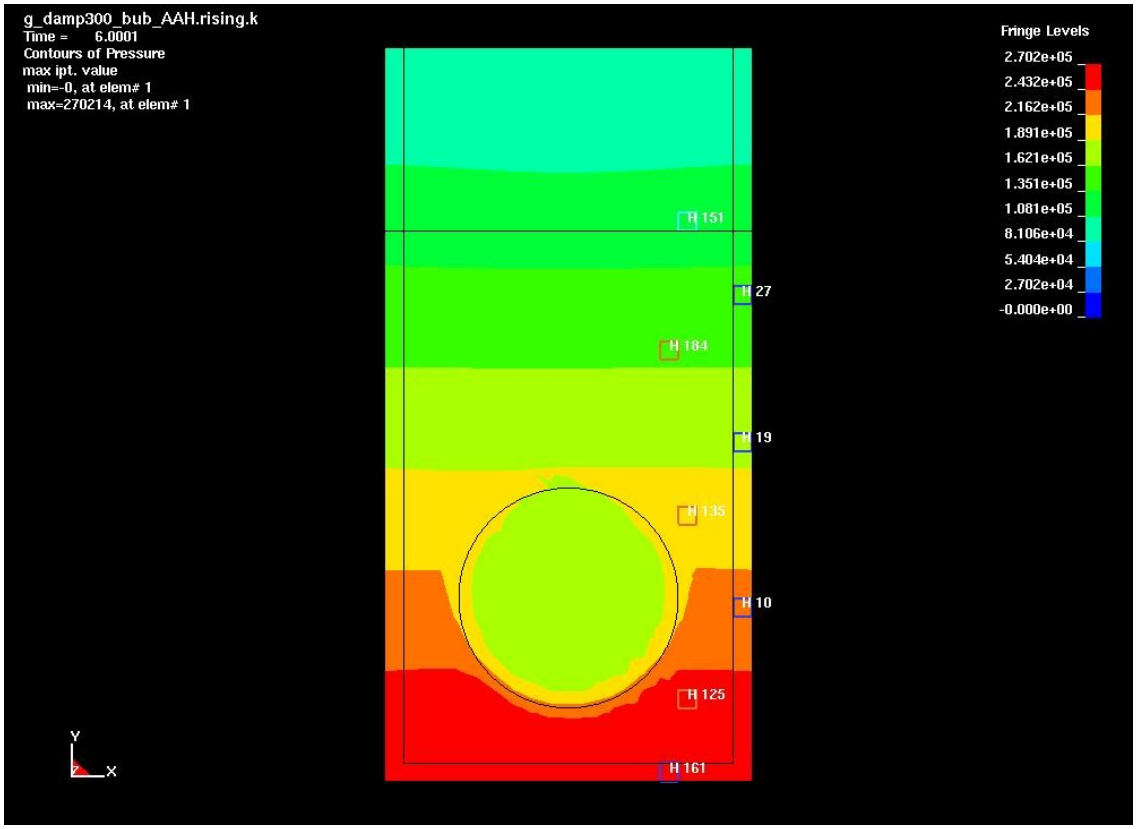


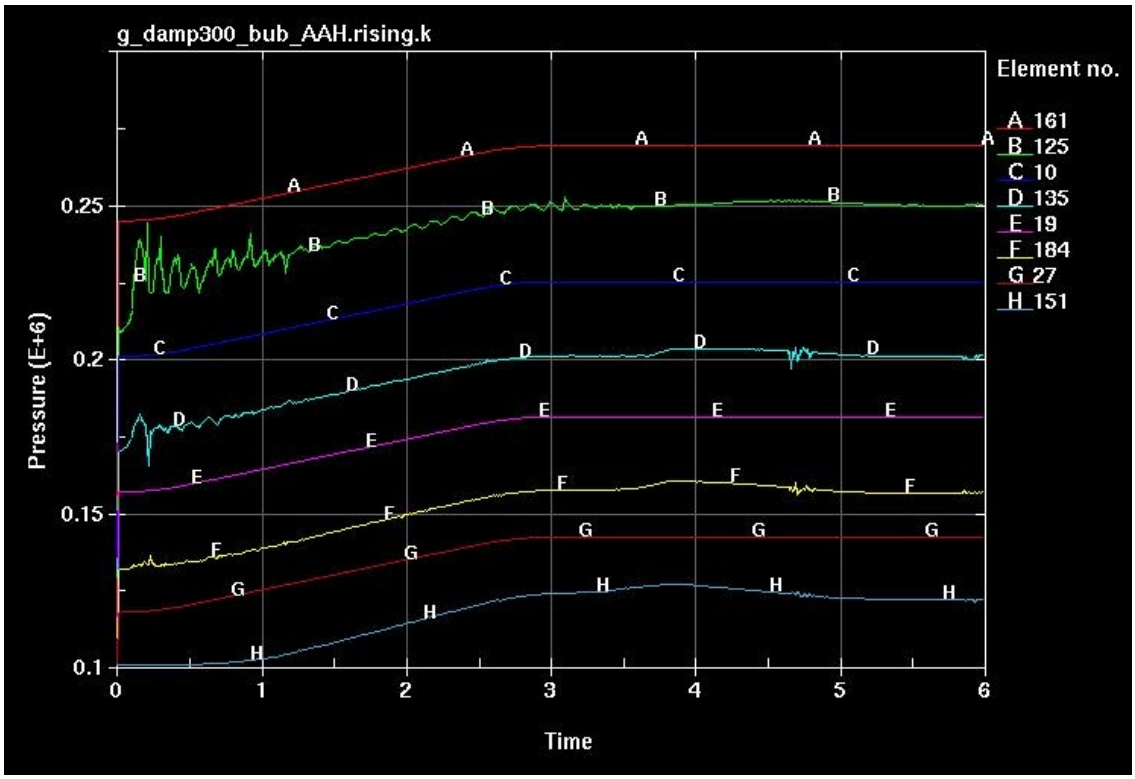
The effect of *ale_ambient_hydrostatic is that, as the water height changes, the pressure in each ambient element changes accordingly. The change in pressure of the ambient elements drive pressure gradients through fluid field causing the water level to change in the main body of the fluid mesh. The change in water level (green material) can be seen in the two figures below.





Figures showing pressure fringes and pressure time histories are provided below. Note the difference in character between the pressure time histories of the ambient elements (smooth, as prescribed by *ale_ambient_hydrostatic) and the pressure time histories of the elements in the nonambient fluid field (not quite as smooth).


g_damp300_bub_AAH.rising.k
Time = 6.0001
Contours of Pressure
max ipt. value
min=-0, at elem# 1
max=270214, at elem# 1
Fringe Levels
2.702e+05
2.432e+05
2.162e+05
1.891e+05
1.621e+05
1.351e+05
1.081e+05
8.106e+04
5.404e+04
2.702e+04
-0.000e+00
H 151
H 27
H 184
H 19
H 135
H 10
H 125
H 161
Y
Z
X


g_damp300_bub_AAH.rising.k
Element no.
A 161
B 125
C 10
D 135
E 19
F 184
G 27
H 151
Pressure (E+6)
0.25
0.2
0.15
0.1
0
1
2
3
4
5
6
Time

Postprocessing Notes

Because ALE materials do not follow the mesh and can migrate from part to part, viewing the location of ALE materials when postprocessing is a little different than viewing Lagrangian materials. Lagrangian material is tracked by part ID. ALE material is tracked by AMMG ID. When postprocessing the d3plot database with LS-PrePost, one can view any of the AMMGs by selecting *Selpar > Fluid*. Each AMMG is then added to the Part Selection list so that it may be selected and viewed just as one might select and view a part.

Alternately, the location of each AMMG can be seen by fringing volume fraction of the respective AMMG. A volume fraction of 0.0 in a region indicates the complete absence of the AMMG in that region. A volume fraction of 1.0 in a region indicates that the region is completely filled with that AMMG. A number between 0.0 and 1.0 means the region is partially filled with that AMMG and thus partially filled with one or more other AMMGs. For example, to see where material associated with AMMG #2 resides, one could click *Fcomp > Misc > volume fraction mat#2*. The space shown in red (corresponding to fringe values between 0.9 and 1.0) is the space exclusively or predominantly occupied by AMMG #2.

Fcomp > Misc > dominant fluid material identifies the various AMMGs by assigning a value of 1.0 to those cells comprised predominantly of AMMG 1, 2.0 to those cells comprised predominantly of AMMG 2, and so on.

History > Element > species mass mat#n plots a time history of the mass of AMMG #n in the selected element. The total mass of the element would be the sum of all the species masses.

In the case of an ALE element, the seven standard history variables written to the d3plot database and labeled as x-stress, y-stress, z-stress, xy-stress, yz-stress, zx-stress, and plastic strain in LS-PrePost are the volume-fraction-weighted values of those respective components. In other words, for an element which has a mix of AMMG 1 and AMMG 2 and which contains no other AMMG, the x-stress in that element will be (element x-stress in AMMG 1)*(volume fraction of AMMG 1) + (element x-stress in AMMG 2)*(volume fraction of AMMG 2).

As with any element, pressure, effective stress (von Mises), etc. are not contained in d3plot but rather are computed by LS-PrePost using the x-stress, y-stress, … , zx-stress values from d3plot. Thus for an ALE element, pressure, effective stress, etc. are calculated based on the volume-fraction-weighted x-stress, y-stress, etc.

In addition to the seven standard history variables, each material model has a particular number of so-called extra history variables which can optionally be written to d3plot. When postprocessing with LS-PrePost, any saved extra history variables are listed under *Fcomp > Misc* and under *History > Element* as “history var#1”, “history var#2”, and so on. Whether dealing with Lagrangian or ALE parts, the number and meaning of the extra history variables for each material model is specific to that material model. To write these extra history variables, NEIPH in *database_extent_binary must be set to the number of extra history variables desired.

The bookkeeping scheme used in writing extra history variables gets complex with multi-material ALE parts are included in the model and is best explained by way of an example.

When there are ALE parts in the model, useful information about the stored extra history variables can be ascertained from the messag file, e.g., the messag file might say…

```
*** Note history variables in lsprepost --> Fcomp -->misc
    history variable      1 to      4 belongs to AMMG      1
    history variable      5 to     14 belongs to AMMG      2
    history variable     15 to     26 belongs to AMMG      3
    history variable     27 to     39 belongs to AMMG      4
```

These statements warrant some explanation. First, know that AMMG 1 is always unique in the sense that the seven standard history variables are not written for AMMG 1. The statement in d3hsp "history variable 1 to 4 belongs to AMMG 1" means that the first four extra history variables in the database are the four extra history variables associated with the material model of AMMG 1. Again, the number and meaning of extra history variables is specific to that particular material model. (Bear in mind that most of the extra history variables for any given material model are of little or no interest to the engineer. Only occasionally will the engineer be interested in extra history variables, e.g., those representing damage and temperature in *mat_015.) The statement "history variable 5 to 14 belongs to AMMG 2" indicates that a total of ten history variables (labeled as history var#5 to history var#14 in LS-PrePost) are written for AMMG 2. There are seven standard history variables written for each AMMG (excepting AMMG 1 as noted above), thus the first seven history variables written for AMMG 2 (history var#5 to history var#11) are these seven standard history variables (x-stress, y-stress ..., plastic strain) for AMMG 2 and the subsequent three (history var#12 to history var#14) are extra history variables associated with the material model of AMMG 2. In like fashion, history var#15 to history var#21 are x-stress, y-stress, … , plastic strain for AMMG 3 and history var#22 to history var#26 are extra history variables associated with the material model of AMMG 3. Finally, history var#27 to history var#33 are x-stress, y-stress, … , plastic strain for AMMG 4 and history var#34 to history var#39 are extra history variables associated with the material model of AMMG 4.

Please note that to actually write 39 extra history variables to the d3plot database (4 for AMMG 1 + 10 for AMMG 2 + 10 for AMMG 3 + 13 for AMMG 4 = 39 total), NEIPH must be set to 39 in *database_extent_binary.

When using *Fcomp* to view history variables in ALE parts, it's sometimes clearer to view isosurfaces rather than fringes. To invoke isosurface viewing, click on the *Frin* button and choose *Isos*.

Aside from the d3plot database, there are other output files which can be useful in postprocessing ALE simulations.

- dbfsi is an ASCII output file created when *database_fsi is included in the input deck. The dbfsi file reports time histories of coupling forces applied to the Lagrangian surface identified on Card 2 of *database_fsi. These coupling forces arise from the penalty-based coupling defined in *constrained_lagrange_in_solid. The dbfsi file also reports coupling pressure, defined as the net, normal coupling force divided by the Lagrangian segment area, and mass flux (mflux), defined as the accumulated ALE material mass moving past a set of segments. Mflux can be used to track leakage if the segments are Lagrangian or advection if the segments are ALE. Mflux is also useful for tracking the mass of material exiting a vent hole.
- dbsensor is an ASCII output file containing time histories of pressure at sensors located in the ALE mesh. The command *database_fsi_sensor defines the sensors and specifies the output time interval. Each pressure sensor follows along with a Lagrangian segment (this segment must be included in the slave side of a *constrained_lagrange_in_solid coupling definition). Optionally, the sensor may be offset a specified distance from the Lagrangian segment in the normal direction.
- trhist is an ASCII output file containing stress and pressure time histories and is created when database_tracer is included in the input deck. A tracer is very similar to a sensor but a tracer does not follow a Lagrangian segment. A tracer can be fixed in space or else made to follow the ALE material.
- fsifor is a binary database created when the print flag is set in field 3, Card 4 of *constrained_lagrange_in_sold and a database name is given on the LS-DYNA execution line using "h=*dbasename*". The binary database can be read into LS-PrePost and enable fringe plots of coupling pressure on the Lagrangian surface, as well as time history plots of coupling pressure and forces on segments.

Troubleshooting

Experience has shown that simulations that include ALE elements commonly need to have the time step scale factor (TSSFAC in *control_timestep) reduced from the default of 0.9 to somewhere around 0.6 in order to maintain a stable solution.

Excessive leakage of ALE material across Lagrangian surface:
See discussion of coupling above.

Negative sliding energy attributable to coupling:
Energy associated with *constrained_lagrange_in_solid coupling is calculated and included in "sliding interface energy" reported to glstat. This coupling energy is not, however, reported to sleout. Thus, the difference between sliding interface energy in glstat and total sliding energy in sleout is the coupling energy. If the coupling energy is negative and exhibits a significant influence on the overall energy balance, i.e., the energy ratio drops due to the negative coupling energy, possible improvements may be seen by (1) refining the ALE mesh, (2) reducing the time step scale factor, and/or (3) reducing FRCMIN in *constrained_lagrange_in_solid, e.g., from 0.5 to 0.25.

Negative volume in advection:

This error can occur when the amount of material moving in/out of an ALE element during the advection is excessive. The possibility of this happening is diminished when NADV is set to 1 in *control_ale (recommended). If the error still occurs when NADV=1, consider the following possible remedies:

- Reduce the time step scale factor TSSFAC.
- Refine the mesh (which in itself may reduce the time step) and, if any ALE elements are badly shaped, improve the mesh quality. Often, a regular ALE mesh with cubical or nearly cubical elements is a good starting point.
- Check material input, especially the material density.
- Use appropriate boundary conditions on the exterior of the ALE mesh. Such boundary conditions would usually consist of either (a) nodal constraints to prevent material from exiting the mesh, or (b) pressure applied on the exterior segments to resist egress of material. The former can be applied manually using *boundary_spc or automatically using EBC in *control_ale. The latter (pressure boundary condition) can be applied manually using *load_segment or automatically using PREF in *control_ale.
- Excessive/inappropriate coupling forces could trigger high material velocity. Review the coupling input, making sure that the Lagrangian segment normal directions are appropriate and that the coupling stiffness is not excessive.

Time step drops dramatically:
This problem is often the result of too little mass in an element, perhaps due to excessive material velocity. If the element controlling the time step is near the exterior of the ALE mesh, see the bullet above on appropriate boundary conditions.