Sandia National Laboratories

Operated for the U.S. Department of Energy by
Sandia Corporation

Albuquerque, New Mexico 87185- 0834

date: July 31, 2005

to: Distribution

from: Randy Schunk, MS 0834, 9114;

subject: Liquid Drop Impact on a Porous Substrate: a level-set tutorial (GT-028.0)

keywords: Level Set, Interface Tracking, Porous media

input records: DARCY_CONTINUOUS, POROUS_LIQ_PRESSURE_FILL, VELO_SLIP_LS

Introduction

The purpose of this tutorial is to help familiarize the user with Goma's coupled continuous fluid/porous-flow coupling capability. The test problem we deploy here was defined by Procter and Gamble through the CRMPC CRADA. A couple of constraints that we have been working with should be mentioned:

-This capability is desired in 3D, and hence our insistence on linear elements and pressure stabilization.

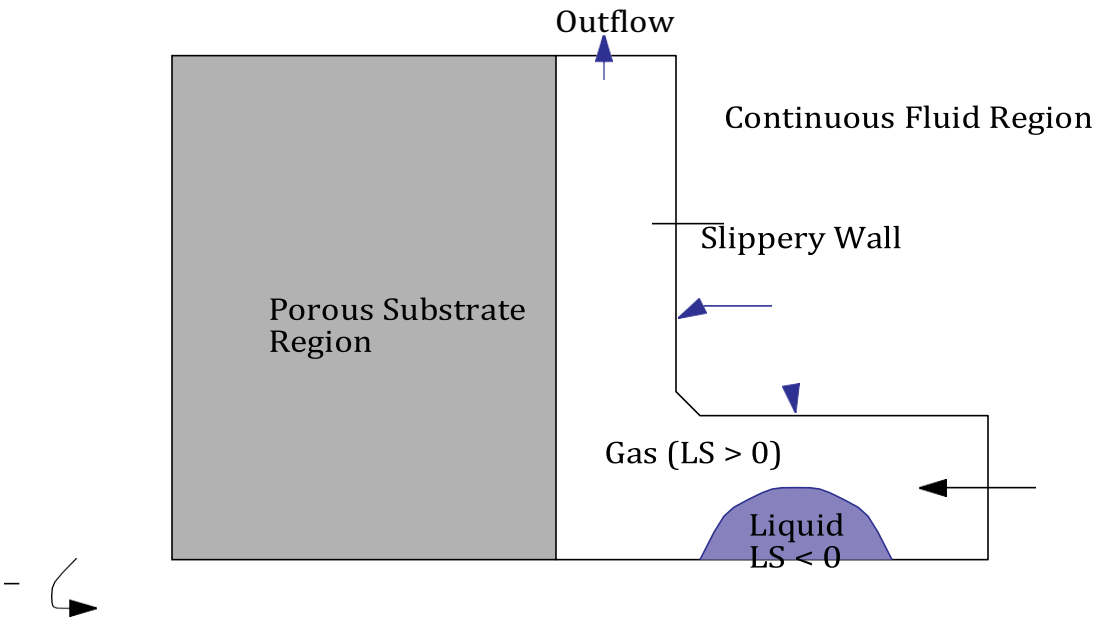
-Run both sub-grid and subelement integration to check for robustness.

It is assumed that those who will run this class of problem with Goma will be considerably experienced in running Goma's continuous fluid capability with level-set front tracking. It also helps to be experienced with Goma's partially saturated flow capability (see GT-009.3 for a tutorial). The most challenging aspects of this problem are the time scale selection (viz. time scale of imbibition and the time scale associated with the Courant limit for the explicit time integrator Goma uses for the level set equation) and renormalization frequency. Also, as in all open flow situations, we find that open flow boundaries must be placed as far away as possible from the zero-level-set contour to avoid "solution pollution" in the problem.

It is also important to mention that the details of this problem make it very difficult (viz. multiple time scales, low Capillary number, coupled physics of porous flow and two-phase continuous flow, etc). We do our best to help guide the user in making solver parameter selections and boundary condition selections, knowing full well there are multiple choices.

Problem Description

The problem consists of two material regions: the porous substrate and the continuous fluid above it. The geometry is depicted in the figure below. The continuous fluid region itself contains two phases, one being the liquid of interest that will be impacting and imbibing into the porous substrate, and the other being the displaced gas. All input parameters for the level set definition correspond to the liquid being represented by the negative level set values, and the gas positive values. With repect to the porous media portion of the problem we refer the reader to GT009.3. Here we will focus our attention



on the intefacial boundary conditions between the two regions.

The template files (viz. input file, Cubit geometry file, material files, etc.) are found in a directory entitled **`PG_drop_porous_tutorial`**. Of course it is desirable to run this problem to the point when all the liquid has imbibed in the porous substrate. A couple of notes (lessons learned) are in order:

-We found that we cannot make the outer wall (labeled “slippery”) an open flow boundary because the uncertainty of the transient flow field creates wiggles in the free surface solution. In fact we observe this phenomenon with all related problems. Confining the flow and keeping the base flow as uniform as possible helps these types of problems. In other tests we did move the “slippery wall” to a much larger distancefrom the action (impact) zone and were able to solve the problem.

-We imposed an inflow velocity of 0.1 mm/msec (or 0.1 m/s) to help deliver the liquid drop to the substrate.

-We find that regularizing the outflow plane (sideset 500) with VELO_TANGENT, thus keeping it to a fully developed profile, again helps keep the wiggles from sneaking into the level set solution.

-We are using a rather esoteric set of units. The base parameter values in the "defs" file are input in CGS, but are then converted to mm-grams-milliseconds. No particular reason for doing this other than it seems to scale the problem better in time and allows for more robust runs. We have run this problem in true CGS units successfully as well.

The mesh is generated by running cubit with **`cap_tube.jou`**. Note that the axis of symmetry is aligned with the x-axis, which is the standard convention with Goma. Also note that this is a linear, 4-node quad mesh (8-node hexes if 3D). This is required for this class of problems as the porous flow formulation demands a Q1 pressure variables. This forces us to use Q1 velocities in the continuous fluid phase and Q1 pressures. Note that this is different than the suggested Q2-Q1_XV velocity-pressure element that is advocated for most level-set problems.

The most important sideset in this problem is **`ss 100`**, which represents the interface between the porous material and the gas/liquid phase. Boundary conditions at that interface must set a variety of conditions. Specifically:

--Continuity of mass flux, viz. the mass that is lost from the liquid in the continuous phase must appear as a mass gain in the porous phase. This condition is DARCY_CONTINUOUS (more below)

--Continuity of pressure: viz. on saturated regions the liquid pressure should be equivalent to the liquid phase pressure in the porous-liquid phase. This is tricky, however, because that pressure in the porous liquid phase should correspond to the saturated portion of the capillary pressure/saturation curve. We will discuss this more below. The boundary condition that enforces this condition is POROUS_LIQ_PRESSURE_FILL.

--Slippage in liquid for advancement of the contact line on the porous surface. This condition is accommodated in VELO_SLIP_LS.

Now some details regarding the base input file, **`input_axi`**. First lets start with the problem description section.

```
Number of processors              = 1
Output Level                      = 0
Debug                             = 0
$Initial Guess                     = read_exoII_file restart.exoII
Initial Guess                     = zero
$Initialize = VELOCITY1 0 -0.8
```

The one card that we should point out here is the **`Initial Guess`** card. Often times it is advantageous to start with an initial guess (restart) that corresponds to a time plane just before drop contact. The most difficult part of the problem is contact and imbibition, and often times you are experimenting with different parameters on the boundary conditions. You can pull a good restart time plane out of a previous run with the SEAMS tool **`algebra`**. When you restart, you also need to make some other changes.

Using the unix 'diff" command for files **input_axi** and **input_axi_restart** yields the following changes:

```
< $Initial Guess                    = read_exoII_file restart.exoII
< Initial Guess                     = zero
---
> Initial Guess                     = read_exoII_file restart.exoII
> $Initial Guess                    = zero
27c27
< Initial Time                      = 0.
---
> Initial Time                      = 10.
44c44
< Level Set Initialization Method = Surfaces 1
---
> $Level Set Initialization Method = Surfaces 1
46c46
< $Level Set Initialization Method  = Exodus
---
> Level Set Initialization Method  = Exodus
```

You can see that our initialization of the level set is based on an exodus II file, rather than the circle entity.

Consider now the time step parameters:

```
Time Integration Specifications
---- ----------- --------------
Time integration                    = transient
delta_t             = 1.0e-3
Initial Time            = 0.
Minimum time step           = 1.0e-10
Maximum time step           = 0.1
Time step parameter                 = 0.
Time step error   = -10.2  0   1   1   1   0   1 0
Printing Frequency                  = 2
Fill Weight Function          = Explicit
Courant Number Limit                = 0.1
```

Within this group of cards a few details are noteworthy. First, with respect to the time stepping, it is important to set reasonable time step intervals, and a maximum time step interval. Several factors weigh into this. The initial time step can be chosen much smaller than the time step you expect to be able to march at, as the automatic time stepping algorithm will increase it up to one of your maxima, or according to the predictor/corrector error, which is set at 10.2 here somewhat arbitrarily. In any case, here are some observations to help you select these numbers:

--Time scales of the problem come from several sources. With respect to the level set field, you should use the explicit fill weight function card as shown, with a courant number limit set to around 0.1. This will help clip your time scale according to a element-based Courant number limit. Now this clipping may be less than your maximum time step.

--On the maximum time step, you should calculate a couple of numbers.

max_delta_t_cfl_gravity = sqrt (length_scale/gravity) = 10 ms.

max_cfl_surface_tension_deltat = sqrt(rho*length_scale^3/sigma) = 200 ms.

These numbers are based on a unit length of 1 mm. If you "aprepro" the "defs" file you will see the numbers from which these calculations are made. Clearly these are much larger than the max value of 0.1 ms which is set in the input deck. You will notice the calculation will cycle between 0.1 ms time steps and the Courant limit which fluctuates.

--Keep in mind that there is another time scale, and that is for the imbibition. This is typically much smaller than that for the capillary free surface. In fact in this exercise you will notice that the time step at impact needs to be drastically reduced to capture the event. The predictor/corrector may crank down on the delta_t for that if you set it smaller.

--If you set the max time step to 200 ms, then you will notice large errors as you will exceed your Courant condition limit. You also have to be aware of the predictor/ corrector error.

--Also be aware that your renormalization tolerance is a factor as well in selecting the proper time stepping. Too many renormalizations translates to significant mass loss. You need just enough to keep your level-set field smooth.

Now consider the Level-Set specifications:

```
Level Set Interface Tracking     = ON
Level Set Subelement Integration = OFF
Level Set Subgrid Integration Depth = 2
Level Set Length Scale           = {LSLS=0.2}
$Level Set Initialization Method  = Surfaces 1
    SURF = PLANE -1.0 0.0 0.0 -0.
Level Set Initialization Method = Surfaces 1
                            SURF = CIRCLE -0.5 -0.0 1.0
$Level Set Initialization Method  = Exodus

Force Initial Level Set Renormalization = yes
Level Set Renormalization Method = Huygens_Constrained
Level Set Renormalization Tolerance = 0.7
Level Set Renormalization Frequency = 1000000
Level Set Reconstruction Method  = POINTS
```

In the first test we will run with a diffuse interface (note that **`Subelement Integration`** is turned off). Subgrid integration is performed with an integration depth of 2. We set the level set length scale to 0.2. This is a critical number, and in this case this size corresponds to twice the average element dimension in the fluid phase. Several boundary conditions actually use this length scale, as discussed below. This parameter may be one you want to adjust depending on how satisfied with the results you are.

The **Level Set Renormalization Tolerance** you will note is set to 0.7. This was "happened upon" by trial-and-error, and as we will show is probably too large once the impact event occurs. We lowered it to 0.2 or so and restarted the impact even so as to keep the results smoother. The remaining level set parameters are more-or-less standard. Details can be obtained by consulting the Goma user-manual (version later than Rev. 4.0) or consulting the level-set tutorial GT-020.3.

The boundary conditions are now discussed:

```
Boundary Condition Specifications
-------- --------- --------------
Number of BC                      = -1
#######################################
## Level Set BC's

# Interface of porous & fluid

BC = POROUS_LIQ_PRESSURE_FILL SS 100 1 2 {-pcmin} {-pcmax} {3.0*LSLS}
BC = DARCY_CONTINUOUS SS 100 1 2
/* Notice the one-sided application of this */

$Must have this no-slippage on the porous-continuous interface.
BC = VELO_SLIP_LS SS 100 0.04 10000. 0.0 0.0 0.0 1.e-6

# External wall for slip
BC = VELO_NORMAL SS 150 0.0
# Free Surface
BC = LS_CAPILLARY LS 0

# Entrance BC
BC = U NS 200 -0.1
BC = V NS 200 0.0

#Exit BC
BC = VELO_TANGENT SS 500 0 0. 0. 0.

## Symmetry Plane BC's
BC = V NS 400 0.0
END OF BC
```

The first three boundary conditions are critical to this problem, with the rest being more-or-less standard for the seasoned Goma user. Please note the issues mentioned above with open flow boundaries, and the reasons for the VELO_TANGENT outflow plane and VELO_NORMAL slippage on the walls.

We discuss the first three BCs in the following figure. Noteworthy is that these BCs depend on several length scales, some of which are related as multiples of of level-set length scale. For sharp interface methods, such as in the subelement integration strategy, there is no level-set length scale, so the boundary conditions are often "mushed" out over another chosen length scale which is included in the boundary condition card. DARCY_CONTINUOUS basically applies the "imbibition" boundary

condition to the continuous fluid equations to allow for liquid to leave the continuous fluid domain and enter the porous media domain. This boundary condition uses a length scale to turn this condition on, viz. a length scale less than which "contact" is declared. VELO_SLIP_LS uses a length scale which characterizes the width of the "mushy zone" and applies a slip/no-slip condition to the liquid phase. In the gas phase it is furbished to allow for complete slip at the solid/fluid inteface when the liquid is approaching contact, to help force the gas from the contact zone. Please see GT-029.1 for a further explanation. Finally, POROUS_LIQ_PRESSURE sets the porous-liquid phase pressure to the saturated value where contact is declared, and to the unsatured "dry" value where it is not. The length scale tied to that condition characterizes the transition region. Keep in mind that these parameters are necessitated by our inability to model the true mechanisms of slip, contact, etc. at a molecular level. Without that fidelity, such parameters are unavoidable. You must experiment with these length scales in order to get your problem to run. Typically we start with 1/2 the element size at the interface, and increase from there. Note that in some cases Goma has preset a multiplier on these length scales for purposes of applying the condition, viz. typically the user tends to set these to one or two widths of an interfacial element, whereas experimentation has found that many more widths are required to get the desired behavior. We realize this is ad hoc and hope everyone understands that it is necessary.

Note that we are not imposing a wetting contact angle. We have found that WETTING_SPEED_LINEAR confounds the system of equations at the interface as we have already overloaded the liquid momentum equations with conditions on slip (VELO_SLIP_LS) and conditions on penetration (DARCY_CONTINUOUS). We've no degrees-of-freedom left to impose a contact angle. If it is an imbibing media (that is, the saturation curve characteristics favor the liquid phase as the wetting phase), then the contact angle is determined by this balance of forces and likely it will be nearly perfect wetting.

Note that in the problem description section of the input deck, we have one equation being solved in the porous material, viz. that for the liquid phase pressure in a partially saturated porous media. This is actually a Darcy equation, with the gas pressure taken as constant. We have the Navier Stokes system and the level-set equation being solved in the bulk of the fluid material.

DARCY_CONTINUOUS

BC = DARCY_CONTINUOUS SS # {Solid_Mat_Id} {Liquid_Mat_ID} {length_scale}

$$\rho \underset{\sim}{n} \cdot (\underset{\sim}{v} - \underset{\sim}{v}_s) = \rho_l n \cdot v_{darcy}$$

Kinematic condition for porous-fluid inteface. Left hand side is activated once a specified length scale gap is passed. For diffuse interfaces it is 4.*LSLS. For sharp interfaces it is specified as the four times the optional float {length_scale}.

VELO_SLIP_LS

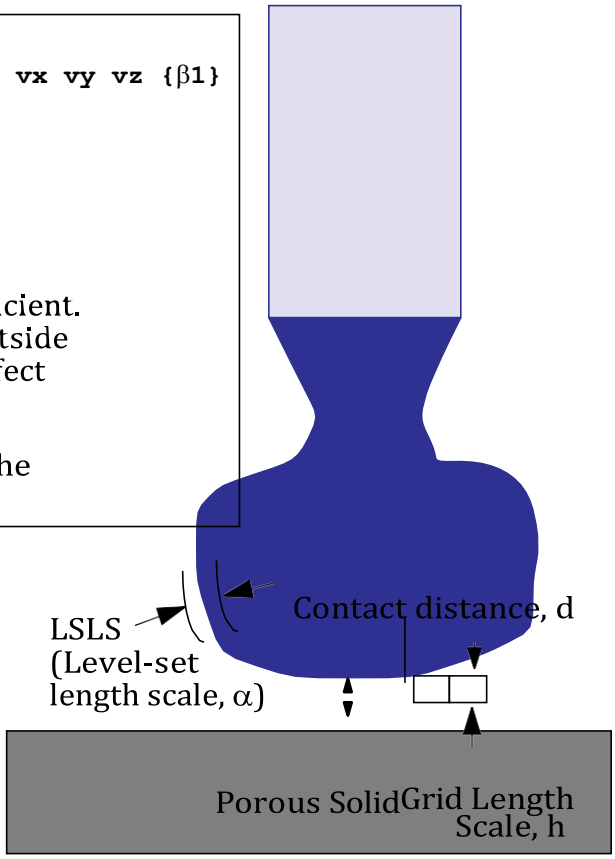
BC = VELO_SLIP_LS SS # {slip_width} {β0} vx vy vz {β1}

$$\frac{1}{\beta} \underset{\sim}{v} - \underset{\sim}{v}_s) = \underset{\sim}{n} \cdot \underset{\sim}{T}$$

$$\beta \equiv (\beta_0 - \beta_1)\delta(\alpha)\alpha + \beta_1$$

Navier Slip Condition with variable slip coefficient. Typically β1 is set small for no slip (active outside the delta function) and β2 is set large for perfect slip.

Note: β is overridden to a large value if in the gas and d < 8*slip_width



POROUS_LIQ_PRESSURE_FILL

BC = POROUS_LIQ_PRESSURE_FILL SS #
{Solid_Mat_Id} {Liquid_Mat_ID} {-pcmin} {-pcmax} {length_scale}

basically sets the Liquid-phase porous pressure to the Saturated value if full contact is made, and to the unsaturated ("dry" value) where no contact is present.

```
Number of Materials = 2
MAT = porous    1

        Coordinate System = CYLINDRICAL
        Element Mapping   = isoparametric
        Mesh Motion = LAGRANGIAN
        Number of bulk species = 0
    Number of EQ = 1
         EQ = porous_unsat Q1 P_LIQ Q1     1.   1.   1.   1.   1.

MAT = liquid_gas      2
  Coordinate System = CYLINDRICAL
  Element Mapping   = isoparametric
  Mesh Motion = ARBITRARY
  Number of bulk species = 0
   Number of EQ = 4
    EQ = momentum1   Q1 U1 Q1        1.   1.   1.   1.   1.  0.
    EQ = momentum2   Q1 U2 Q1        1.   1.   1.   1.   1.  0.
    EQ = continuity  Q1_XV P  Q1_XV       1.                      0.
   EQ = level_set   Q1 F  Q1        1 1     1
```

Only additional feature noteworthy here is the basis function used for the interpolation of the pressure in the continuity equation. Namely, we use **`Q1_XV`** discontinuity capturing for the capillary pressure at the interface. This is required for good mass conservation in axisymmetric problems (**`viz. Coordinate System = CYLINDRICAL`**). Also note that we need to deploy pressure stabilization because we are using equal order interpolation.

Now consider the porous media properties that we will vary in this testing(**`porous.mat`** with aprepro variables defined in '**`defs`**' file).

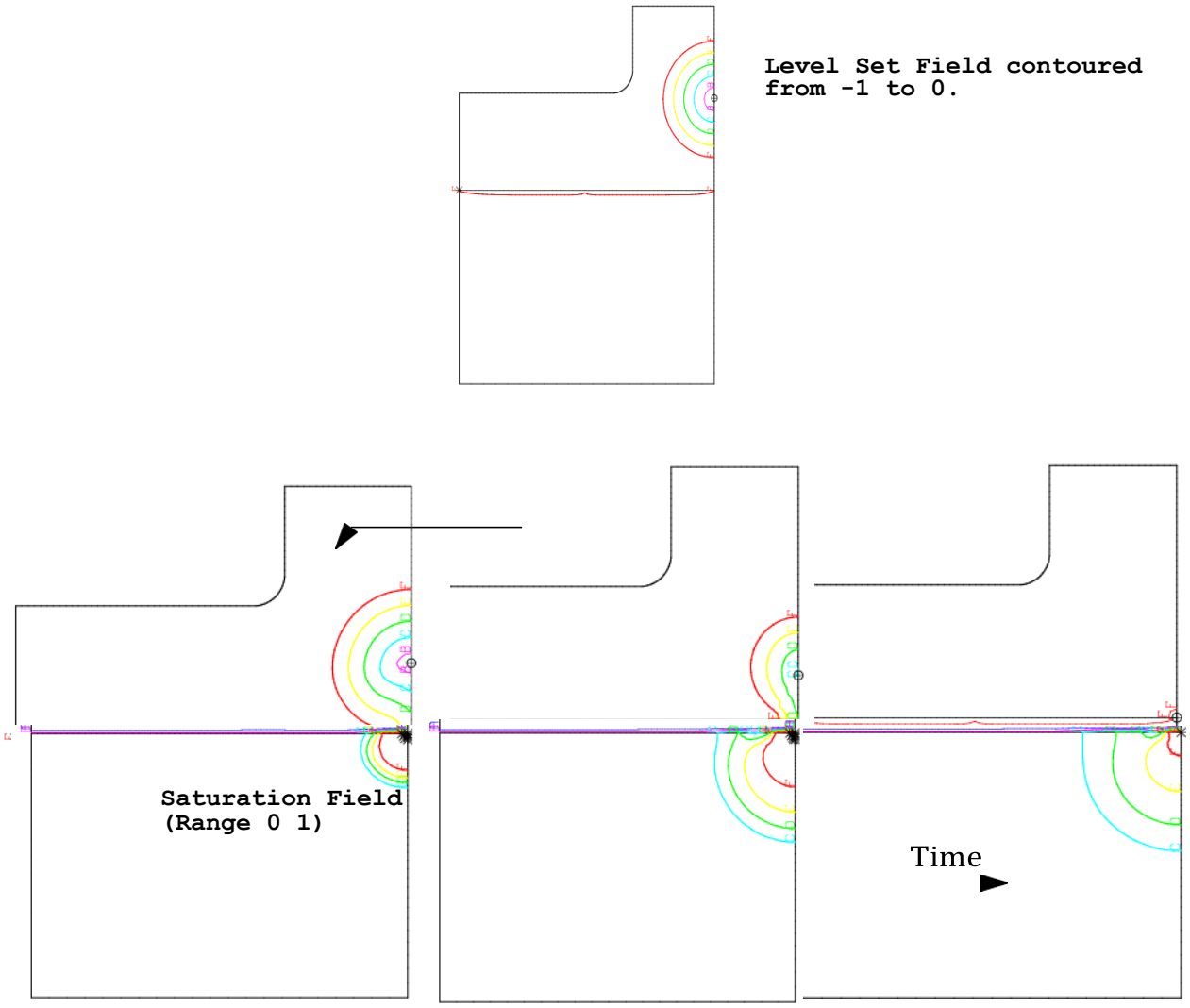
```
---Microstructure Properties (Porous section 1)
...
Permeability = CONSTANT{permeab/100.}
...
$$Material 1
Saturation               = TANH 0.2 0.015 5.33 0.55
$$Material 2
$$Saturation                 = TANH 0.1163 0.0 2.4 0.005
```

Here we will change the permeability of the substrate and the saturation vs. capillary pressure curve and see what happens to the flow. The parameters were chosen to mimic real materials. Goma tutorial GT-009.3 should be consulted for more information.

With respect to the fluid material file, everything is standard, with the viscosity, density and gravity all using the "Second Level Set" property capability. GOMA tutorials GT-009.3 and GT-020.2 should be consulted for further descriptions.

Results.

First run **input_axi** with the permeability set to 10 times less than the standard (see Permeability card in **porous.mat**). Results are in **highperm.exoII**. Note the dominance of capillary suction. The sample results in the following figure refer to this case. Note that:Imbibition is so dominant that it sets the shape of the drop.



Second, run Goma with the permeability set to 100 times less than the standard. In this case you should be able to run from the **restart.exoII** file using **input_axi_restart**. Check to see if the restart file has an "impending" collision all queued up. If not, run **input_axi**. In the material file you will simply change

```
Permeability  = CONSTANT  {permeab/10.}
```

to

```
Permeability  = CONSTANT  {permeab/100.}
```

Here we have included an mpeg move file for this run, **`lowperm.mpg`** Note greater resistance to suction leads to recoil of drop. With this lower permeability note that the drop imbibition is impeded enough that the drop undergos oscillations from impact as it is drawn into the substrate.

Third run is with permeability=permeab/100 and Saturation function for material 2. Note that the time scale for imbibition is much greater, viz. the drop impacts and sits on the substrate. At time=125 ms we stopped and restarted the calculation with much larger time steps to see if we can follow the rest of the calculation. We found we had to increase the Courant number limit and the renormalization tolerance (please diff **`input_axi_restart`** and **`input_axi_restart_1`** for exact values).

All of these runs thus far deploy subgrid integration and so-called "diffuse" interface models. You can also run these problems with a sharp interface representation, with the so-called "Subelement Integration" capability. The proper settings for subelement integration are in the Goma input file **`input_axi_subelement`**. The main difference is the "darcy_length" parameter that is added to the DARCY_CONTINUOUS card. This is required to set the length scale on which a decision to apply the DARCY "suction" to the porous/fluid interface. In the diffuse interface case this length scale is set to four times the level-set length scale. Because there is no level-set length scale for the sharp, subelement case, we must specify one through the optional float on DARCY_CONTINUOUS:

```
> BC = DARCY_CONTINUOUS SS 100 1 2 {darcy_length = 0.04 }
```

Also note that you must set the level-set length scale to zero and turn off sugrid integration. Using Unix "**`diff`**" utility on files **`input_axi`** and **`input_axi_subelement`**to see the necessary changes. The performance per time step is slightly better is slightly better, but the calculation does not seem any more robust. In fact, at the time of this memo revision, Huygens Renormalization that is required for axisymmtric cases was not working for Subelement Integration. Hence, subelement integration is not recommended.

Trouble Shooting Tips:

When we began setting up this problem by a conversion from 2D to 2D axisymmetric, we noted the following:

Noticed a shrinking drop as the calculation proceeds. Classic mass-loss problem with LS technology. Here are some notes.
Running with Max delta_T = 1.0 and no explicit: bubble disappears.

Running with Max delta_T = 0.1 and explicit with Courant=0.1. Still shrinks.

Run with gravity and Huygens_Constrained. Bombs and NaNs.

Turned out that you must run Q1_XV Pressure interpolation, viz. a Q1-Q1_XV velocity-pressure element if you must you Q1s. This fixed the mass loss for an axisymmetric problem. I noticed that if you ran Q1Q1 for cartesian, the mass loss was not that bad and manageable with renormalization and time-step control and refinement.

Cannot use WETTING_SPEED_LINEAR with DARCY_CONTINUOUS or VELO_SLIP_LS because of bad results. Too many constraints on the momentum equations.

When running Subelement integration you must use Huygens renormalization and set level-set length scale to 0. N.B. be careful to set the aprepro variable LSLS to a nonzero value, e.g. {LSLS = 0.2}