
Tutorial: Horizontal Film Boiling

Purpose

If the wall temperature is much higher than the saturation temperature of the liquid in contact with the wall, then the entire wall surface is immersed in vapor. Due to the boiling mass exchange occurring at the vapor-liquid interface, bubbles of gas are periodically produced and emitted upwards. Such a regime is known as film boiling. The purpose of this tutorial is to provide guidelines and recommendations for setting up and solving the applications, including advice on mesh resolution and solver settings.

Note: *In this tutorial, we have to compile the UDFs, as they cannot be run through the FLUENT interpreter for this version of FLUENT.*

Prerequisites

This tutorial assumes that you are familiar with the FLUENT interface, and have a good understanding of basic setup and solution procedures. This tutorial will not cover the mechanics of using the models; instead, it will focus on the application of the models. If you have not used these models before, the FLUENT Tutorial Guide will provide you with the necessary experience.

Problem Description

The problem to be solved in this tutorial is shown in Figure 1.

The wall has a temperature 10K above saturation. Initially, the linear temperature profile (from T_{wall} to T_{sat} in the positive Y direction) is patched on the liquid domain. Note that the double precision version of FLUENT is used to solve this problem.

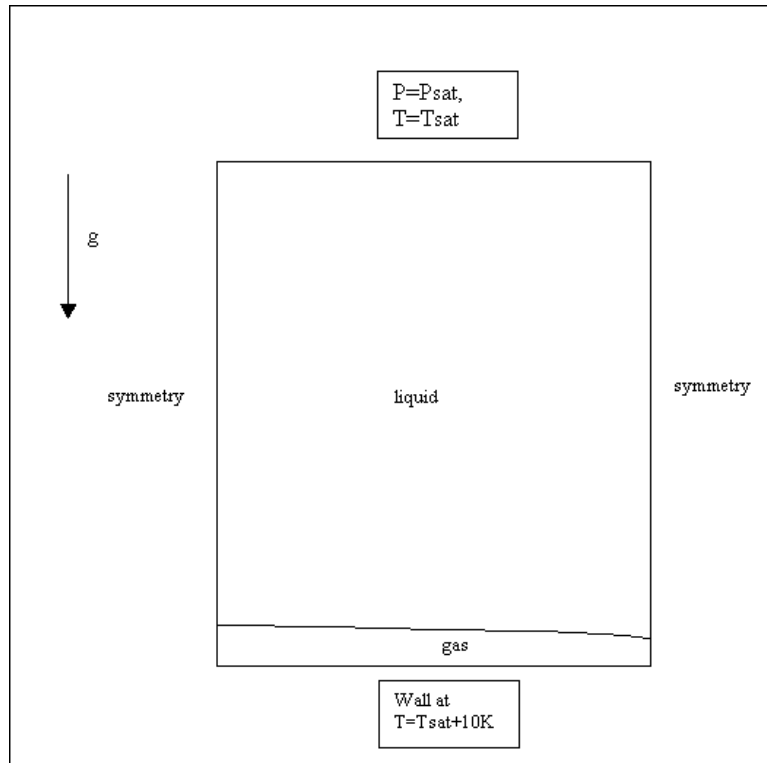


Figure 1: Schematic of the Problem

Setup and Solution

Step 1: Grid

1. Read the grid file (`test-2d.msh`).
2. Check and display the grid (Figure 2).
3. Scale the grid.

Step 2: Models

1. Define the solver settings.
 - (a) Turn on the Volume of Fluid multiphase model and accept the default settings.
 - (b) Enable heat transfer by activating the energy equation.

Step 3: Materials

1. Create two new materials, `vapor` and `liquid`, with properties (in SI units) given in the following table:

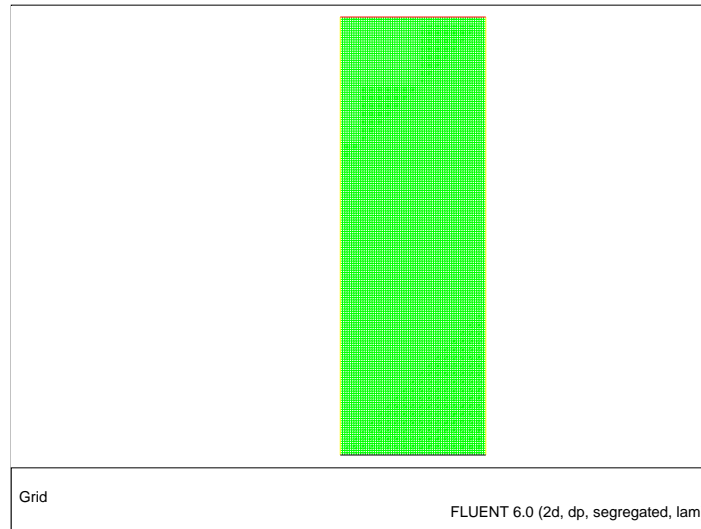


Figure 2: Grid Display

Parameters	liquid	vapor
Density (kg/m^3)	200	5
Specific heat ($j/kg - k$)	400	200
Conductivity ($w/m - k$)	40	1
Viscosity ($kg/m - s$)	0.1	0.005

2. Define the primary (vapor) and secondary (liquid) phases.
 - (a) Specify vapor as the primary phase and enter `gas` as the Name.
 - (b) Specify liquid as the secondary phase and enter `fluid` as the Name.
 - (c) Specify the interphase interaction.
 - i. In the Phase Interaction panel, turn on Surface Tension and enter 0.1 n/m for the Surface Tension Coefficient.

Step 4: Operating Conditions

1. Turn on Gravity.
2. Set the Gravitational Acceleration in the Y direction to -9.81 m/s^2 .
3. Turn on Specified Operating Density and set the Operating Density to 5 kg/m^3 .

Step 5: Compile the the User-Defined Function

In this problem, the user-defined function (UDF) is used to specify mass transfer between phases. If you want to use a mass transfer model that is not available in FLUENT, you can specify it yourself using a UDF. The mass transfer terms for each phase (vapor and liquid) are equal and have opposite signs. They are specified as source terms in volume fraction

equations and have units of $\text{kg}/\text{m}^3/\text{sec}$. In addition, there is an energy source temperature, to take into account the latent heat absorbed/released, which has units in Watts/m^3 and is prescribed for mixture energy equation. For more information on compiled UDFs, refer to the section on Compiled UDFs in Chapter 7: Compiling and Linking of the FLUENT UDF Manual.

1. Compile the UDF, `boiling.c`, using the Compiled UDFs panel.
2. Enter the name of your library directory (e.g., `libudf`) if it is located in your working directory, or the complete path (e.g., `~myhome/myfiles/libudf`).
3. Click **Open** to link your shared library to the FLUENT executable.
4. Set the number of User Defined Memory Locations to 3.

Note: *This step is necessary as the UDF uses three UDMLs.*

5. In the User-Defined Function Hooks panel, select `area_density` in the drop-down list for Adjust Function.

Step 6: Boundary Conditions

1. Set the boundary conditions for fluid.
 - (a) In the Boundary Conditions panel, retain the Phase as mixture and click the **Set...** button.
 - (b) Turn on **Source Terms** and `s udf energy` in the **Energy** drop-down list.
 - (c) In the Boundary Conditions panel, change the Phase to fluid and click the **Set...** button.
 - (d) Turn on **Source Terms** and select `udf liquid` in the drop-down list for **Mass**.
 - (e) In the Boundary Conditions panel, change the Phase to gas and click the **Set...** button.
 - (f) Turn on **Source Terms** and select `udf gas` in the drop-down list for **Mass**.
2. Set the conditions for the wall boundary, heat.
 - (a) In the Boundary Conditions panel, change the Phase to mixture and click the **Set...** button.
 - (b) Under **Thermal Conditions**, select **Temperature** and enter 510 k for **Temperature**.
This indicates 10 k superheat with respect to the saturation temperature 500 k.
3. Set the boundary condition for the pressure-outlet, outlet.
 - (a) Enter 500 k for **Backflow Total Temperature**.
 - (b) This will prevent gas entrance at the outlet when reversed flow occurs and also sets the saturation temperature for the liquid if reversed flow occurs.
 - (c) In the Boundary Conditions panel, change the Phase to fluid and click the **Set...** button.
 - (d) Enter 1 for **Backflow Volume Fraction**.

Step 7: Solution

1. Set the solution parameters.
 - (a) For the Under-Relaxation Factors, set the values of Pressure and Momentum to 0.7, and the rest to 1.
 - (b) Under Discretization, select PRESTO! for Pressure, PISO for Pressure-Velocity Coupling, and Second Order Upwind for Momentum and Energy.
 - (c) Turn off Skewness Correction.
2. Initialize the solution.
 - (a) Initiate with pressure and velocity components at zero.
 - (b) Set the fluid Volume Fraction to 1.
 - (c) Initialize the flow field Temperature to 500 k.
3. Define a custom field function for initializing the temperature profile.
 - (a) Create the custom field function, $510 - y * 10/0.1168$.

This function is needed to initialize temperature profile, ranging from 510 K at $y=0$ (for heat wall) to 500K at $y = 0.1168$ m (for outlet).

Hint: Select y by selecting Grid... and Y-Coordinate in the Field Functions drop-down lists.
4. Patch a temperature using custom-function-0 for the fluid zone.
5. Monitor the solution during iteration.
 - (a) Enable the plotting of residuals.
 - (b) Define a surface monitor.
 - i. Increase the value of Surface Monitors to 1.
 - ii. Turn on Plot, Print, and Write.
 - iii. Under Every, select Time Step and click Define... to open the Define Surface Monitor panel.
 - iv. Select Wall Fluxes... and Surface Nusselt Number in the Report Of drop-down lists.
 - v. Under Surfaces, select heat.
 - vi. In the Report Type drop-down list, select Area-Weighted Average.
 - vii. In the X Axis drop-down list, select Flow Time.
 - viii. Under File Name, enter `nusselt-1.out`.
 - (c) Define a volume monitor.
 - i. Increase the value of Volume Monitors to 1.
 - ii. Turn on Plot, Print, and Write.
 - iii. Under Every, select Time Step and click Define... to open the Define Volume Monitor panel.

- iv. Select Phases... and Volume fraction of gas in the Field Variable drop-down lists.
 - v. Under Cell Zones, select fluid.
 - vi. In the Report Type drop-down list, select Volume-Average.
 - vii. In the X Axis drop-down list, select Flow Time.
 - viii. Under File Name, enter `vol-mon-1.out`.
6. Set Reference Values for Length to 0.0778 m and Temperature to 500 k.
 7. Start the calculation.
 - (a) Set Time Step Size to $2e-4$.
 - (b) Set the Number of Time Steps to 20000.
 - (c) Click Iterate.

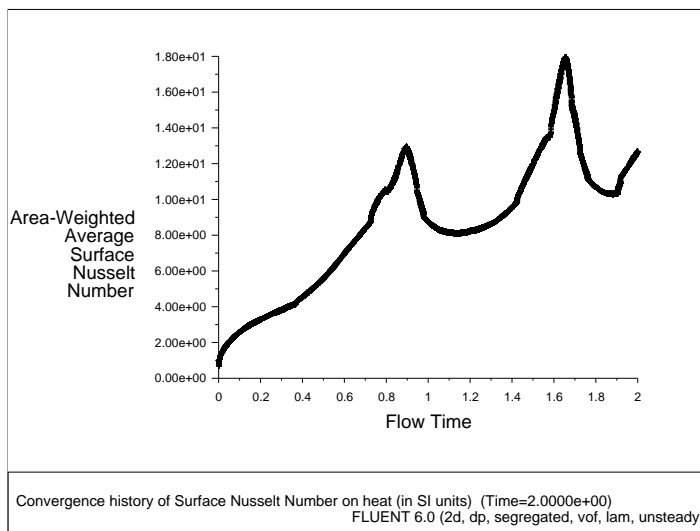


Figure 3: Convergence History of Surface Nusselt Number

Step 8: Postprocessing

1. Display filled contours of volume fraction of gas after 4500 time steps ($t = 0.9$ s).
 - (a) Select Filled under Options.
 - (b) Select Phases... and Volume fraction of gas in the Contours Of drop-down lists and click Display.
2. Display filled contours of volume fraction of gas after 5500 time steps ($t = 1.1$ s).
3. Display filled contours of user-defined memory after 10000 time steps ($t = 2.0$ s).
 - (a) Select User Defined Memory and `udm-1` in the Contours Of drop-down lists and click Display.

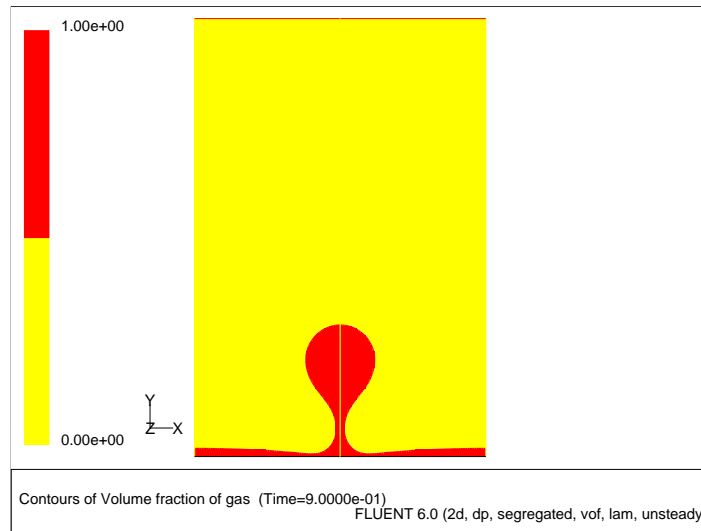


Figure 4: Contours of Gas Volume Fraction at $t = 0.9$ s

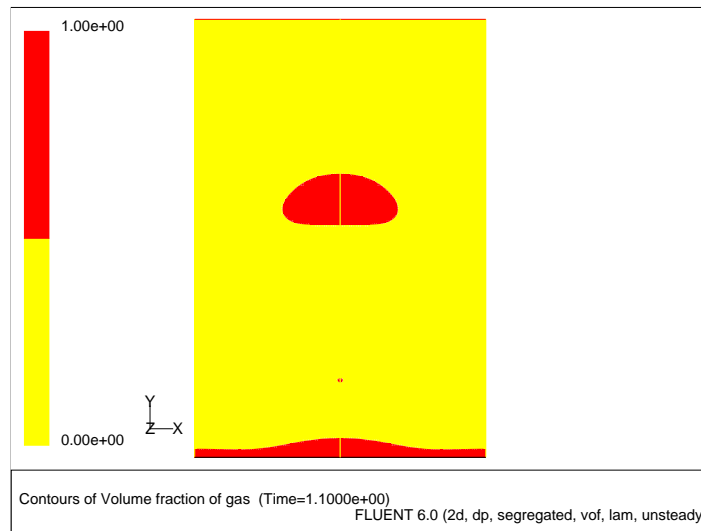


Figure 5: Contours of Gas Volume Fraction at $t = 1.1$ s

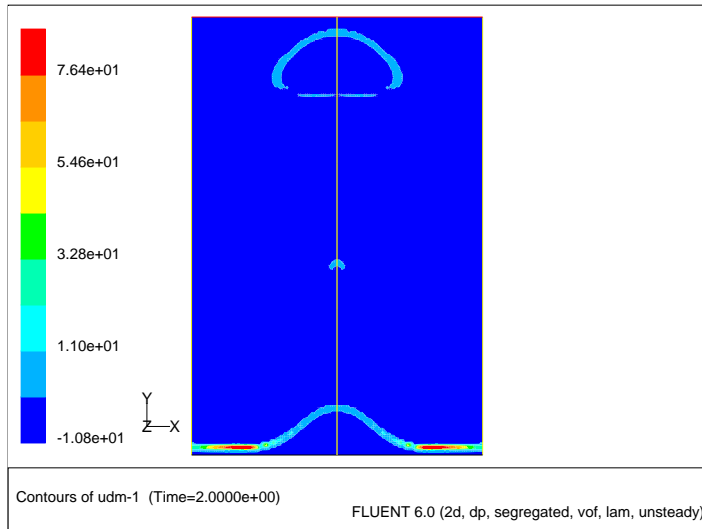


Figure 6: Contours of User-Defined Memory at $t = 2.0$ s

4. Display contours of static temperature after 10000 time steps ($t = 2.0$ s).
 - (a) Select Temperature... and Static Temperature in the Contours Of drop-down lists and click Display.

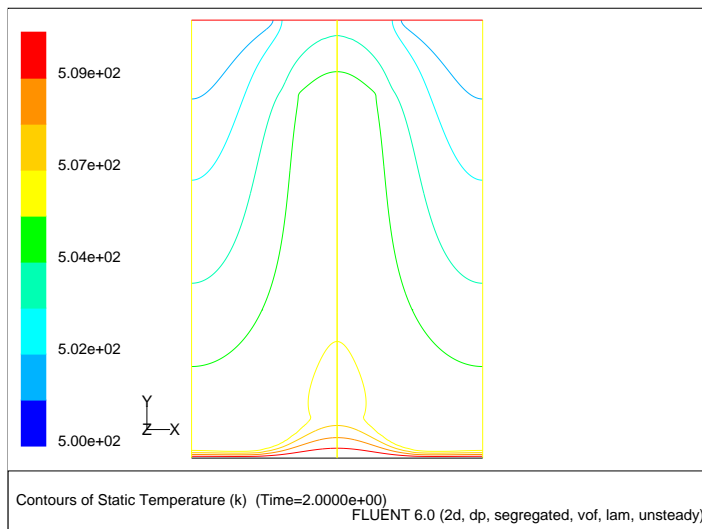


Figure 7: Contours of Static Temperature at $t = 2.0$ s

5. Display contours of volume fraction of fluid after 10000 time steps ($t = 2.0$ s).

Results

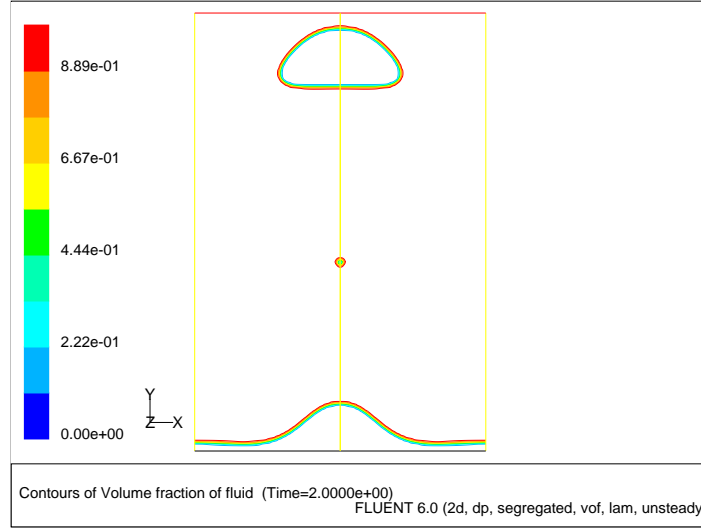


Figure 8: Contours of fluid Volume Fraction at $t = 2.0$ s

This problem requires a UDF for implementing mass exchange source terms in mass equations for each phase (liquid and vapor). Source term in energy equation accounting for latent heat transfer is also required.

The general form of the mass source in the vapor phase is:

$$S_{\alpha_s} = -\frac{(q_g'' - q_l'') \cdot \nabla \alpha_1}{L} \quad (1)$$

Here, q'' is the heat flux across the interface per area of the interface, the subscripts l and g refer to liquid and vapor, respectively, and L is the latent heat.

As a first order approximation, the heat flux difference is represented as:

$$(q_g'' - q_l'') \approx -(\alpha_l k_l + \alpha_g k_g) \nabla T \quad (2)$$

In this approximation, Equation 1 becomes,

$$S_{\alpha_g} = \frac{(k_l \alpha_l + k_g \alpha_g)(\nabla T \cdot \nabla \alpha_l)}{L} \quad (3)$$

Since there is no internal mass sources, mass source for liquid phase becomes:

$$S_{\alpha_l} = S_{\alpha_g} \quad (4)$$

Latent heat source for the energy equation becomes:

$$S_E = -S_{\alpha_g} \cdot L \quad (5)$$

Interfacial properties include surface tension $0.1N/m$, latent heat $1e^5 J/kg$, saturation temperature $T_{sat} = 500K$. The length scale of the problem is the most dangerous wavelength of Taylor-Raleigh instability:

$$\lambda_0 = 2\pi \left(\frac{3\sigma}{(\rho_l - \rho_g)g_y} \right)^{1/2} = 0.0778m \quad (6)$$

The velocity scale of the problem is:

$$\sqrt{g_y \lambda_0} = 0.87m/s \quad (7)$$

Hence the time scale will be:

$$\sqrt{\lambda_0/g_y} = 0.09m/s \quad (8)$$

The domain horizontal width is $\lambda_0/2$ and the vertical height is $3\lambda_0/2$. The mesh resolution is $64(hor) \times 192(ver)$. The initial shape of the vapor-liquid interface must be perturbed to initiate bubble growth. Therefore, there is another initialization UDF which fills with gas all cells satisfying the following condition:

$$y \leq 0.00292 + 0.0006 \cos(2\pi x/\lambda_0) \quad (9)$$

Here $x(y)$ is horizontal (vertical) coordinate in meters.

Nusselt number is an important dimensionless group characterizing boiling heat transfer and is defined as:

$$Nu = \frac{|q''| \lambda_0}{k_l(T_{wall} - T_{sat})} \quad (10)$$

Since the time scale of this problem is 0.01 sec, the time step is $2e-4$, i.e., 50 time steps resolution. In total, the problem should run for about 6000 time steps to capture first bubble emission.

The postprocessing had void fraction and temperature profiles at 4500 and 5500 time steps, corresponding to 0.9 and 1.1 seconds. These times correspond to minimum and maximum thickness of the boiling film at the heated wall. They correspond to minimum and maximum values of Nusselt number, respectively.

Summary

Application of the VOF model in film boiling regime has been demonstrated in this tutorial. Also, UDFs have been used to enhance the standard features of FLUENT.

- `DEFINE_ADJUST` (`area_density`) is a general-purpose macro that can be used to adjust or modify `FLUENT` variables. In this case it calculates dot product and stores the dot product in Equation 3.

For more information, refer to Section 4.2.1 `DEFINE_ADJUST` of the UDF Manual.

- `DEFINE_SOURCE` (`gas`) uses `UDMI(0)` and user input for `L` to compute mass source term (Equation 3) for the gas phase and stores it in `UDMI(1)`. This macro also computes energy source (Equation 5) and stores it in `UDMI(2)`.

For more information, refer to Section 4.3.8 `DEFINE_SOURCE` of the UDF Manual.

- `DEFINE_SOURCE` (`liquid`) assigns `UDMI(1)` with negative sign as a liquid mass source in accordance with Equation 4.
- `DEFINE_SOURCE` (`energy`) assigns `UDMI(2)` as a latent heat source.
- `DEFINE_INIT` (`my_init_function`) initializes gas void fraction in accordance with Equation 7.

For more information, refer to Section 4.2.2 `DEFINE_INIT` of the FLUENT UDF Manual.