Coffee Filter

Comsol Laboration

DN1240, HT12/VT13

1 Physical configuration

On a wet filter falls a drop of coffee every $T = 5$ seconds. We wish to make a movie of how the coffee-concentration in the filter varies during the first 60 seconds.

A coffee filter looks like in Figure 1 and we can let the filter be $h = 2$ mm thick. A drop is considered to contain 1 mmol of coffee and spreads rapidly, on time $T_i = 0.5$ sec, over a circular cylinder, height h, with radius $r = 4$ mm and the center (x_0, y_0), a point of your choice somewhere on the filter.

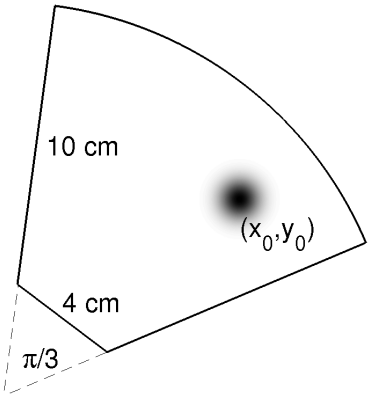

Figure 1: The coffee filter geometry

2 Model

The coffee concentration c evolves in the paper according to the diffusion equation,

$$c_t - \nabla \cdot (D\nabla c) = 0, \tag{1}$$

where $c(x, y, t)$ is the concentration of coffee [mmol/m^3] and D is the Fick's diffusion coefficient [m^2/s]. No flux of coffee out from the filter implies homogeneous Neumann conditions on the boundaries, $n \cdot \nabla c = 0$, where n is the normal to the boundary. To model the drops we add a source term R [mmol/(m^{3}s)] in the right hand side

$$c_t - \nabla \cdot (D\nabla c) = R(x, y, t). \tag{2}$$

The value of D is about 0.0001 [m^2/s]. If you create the geometry in cm, the value will be 0.0001×10^4 [cm^2/s].

2.1 Preparatory analysis

We want to find a formula for $R(x, y, t)$ that well models the generation of drops.

1. Consider first one single drop. Since $R(x, y, t)$ should only affect the dynamics for a short time (T_i) and in a small volume (the cylinder) it means that R must be identically zero for other values of (t, x, y).

   Show that since R should produce 1 mmol it must hold that

   $$\int_0^{T_i} \int_{V_i} Rh \cdot dxdydt = 1,$$

   if we calculate the concentration in mmol / unit volume. (Here V_i is the cross section of the cylinder, i.e. a circular disc.)

2. Let us write $R(t, x, y)$ as a product of a time-dependent function $Q_1(t)$, a space-dependent function $Q_2(x, y)$ and a normalization factor γ. For simplicity we let the functions both be characteristic functions, of $[0, T_i]$ and V_i respectively,

   $$Q_1(t) = \begin{cases} 1, & t \in [0, T_i], \\ 0, & t \notin [0, T_i], \end{cases}, \qquad Q_2(x, y) = \begin{cases} 1, & (x, y) \in V_i, \\ 0, & (x, y) \notin V_i. \end{cases}$$

   Hence, $R(t, x, y) = \gamma Q_1(t) Q_2(x, y)$. Determine the right value for γ.

3. We now consider the full problem where drops are generated every five seconds. Then $Q_1(t)$ should be periodic. To make it easy to implement in COMSOL (see below) we choose the following form

   $$Q_1(t) = \begin{cases} 1, & \sin(\omega t) > b, \\ 0, & \text{otherwise}. \end{cases}$$

   Compute the right values for ω and b!

3 Lab session work

Model: In the `Model Navigator`, select 2D and
`COMSOL Multiphysics/PDE Modes/PDE, Coefficient Form/Time-dependent Analysis`
Geometry: The geometry can be created in cm with a circle of radius 14, two 14 cm squares with a corner at the origin, one rotated 30^o, and one with side 4, and a corner at (0,4), rotated 30^o. Create the filter by appropriate Boolean operations on these figures, under `Draw/Create Composite Object`
Constants and expressions: Define the diffusion coefficient D, time T_i, b, γ, etc. as constants under `Options/Constants`. Note that it is inappropriate to call new quantities t, x, y, c, h, ..., which are already defined in COMSOL.

For the source function $R(t, x, y)$ we use `Subdomain Expressions` under `Options/Expressions`. Recall that in Matlab and COMSOL a Boolean expression is 1 if true and 0 if false. We can therefore define Q_1 and Q_2 as

```
Q1 = sin(omega*t) > b,
Q2 = (x-x0)^2 + (y-y0)^2 < ri^2,
```

and then R as `Q1*Q2*gamma`.

PDE: Set the diffusion coefficient D and source term R (named `c` and `f` respectively) under `Physics/Subdomain Settings`. Also set the initial value to zero under the tab `Init`. Note that you may need to select `Time-dependent` under `Solve/Solver Parameters` first, see below.
Boundary conditions: Set all edges to `Neumann` with `q = g = 0` (the defaults) under `Physics/Boundary Settings`.
Solver: Under `Solve/Solver Parameters` select `Time Dependent` and a time range such as 0:1:60. Run for shorter time, say 10, until it seems to work, this makes the tests go faster. Note that the :1: does NOT control the time-step, only the snapshots saved. You have to set the tolerances under `Solve/Solver Parameters` to change time-steps.
Tasks:

1. Plot the concentration as `Surface` with e.g. `Colormap gray` so it will be darkest where the coffee is strongest. Animate!

   Check that the solution looks reasonable. You may need to manually force the maximum time step to something clearly smaller than the drop creation time T_i. Do this under `Solve/Solver Parameters`, tab `Time Stepping`.

2. Try other values of D (ten times smaller and ten times bigger, for example). Are the changes to the solution as you expect them to be?

3. Plot concentration (`Post Processing/Cross-Section Plot Parameters` and choose `Point plot`) at a point near (x_0, y_0) so you can see if all drops are caught. If not, more stringent tolerance for the time-stepping, and/or max time step.

4. Check by `Subdomain Integration` if all coffee has wound up in the filter! The number of drops and the coffee volume per drop are known. If the results is very wrong, although all drops have been recorded, there is something wrong with your parameter γ. (Be careful here with how the volume scales with h!)

5. Experiment with different mesh refinement, and tolerances for the time stepping. Use the concentration at (x_0, y_0) at a time just after the last drop has diffused for the convergence study. Make a table of the different runs and report the mesh size and time tolerances necessary to obtain $< 5\%$ error

The major source of error comes from the discontinuities in $R(x, y, t)$. Time-stepping is difficult to adjust so that time steps hit exactly the pulse flanks, so we are looking for other ways to reduce errors. Another error is caused by the elements not having edges along the circular edge where $Q_2(x, y)$ jumps. This is fixed as follows:

6. Show first with a 1-D analogy, that the trapezoidal rule quadrature formula in general gives an error proportional to the step size (not squared) on discontinuous integrands. Suggest how the step size could be chosen to recover second order accuracy.

7. To reduce the error in the diffusion problem, add a drop-in circle geometry and make a composite object of the union of circle and filter, maintaining internal boundaries. There is a check box for this. There are two sub-domains, define $R = \gamma Q_1(t)$ in the circle and zero outside. The element edges now fit exactly with the circle and the calculation becomes much more accurate.

8. Investigate how much by comparing with the model in tasks 5 above.