Use of MATLAB m-files in FEMLAB

Dewi L. Kauw

Chem. Eng. 499

Spring, 2003

Supervisor: Prof. Bruce Finlayson

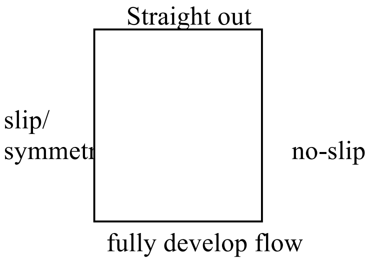
The goal of this project was to predict the pressure drop of a ferrofluid in an oscillating magnetic field. The parameters used were those determined by Kris Schumacher for the polarized magnetic field. A quisi-3D model with magnetic field ramping up, staying steady, and then declining will be solved. The objective was to find out how much difference ramping makes compared with a constant value.

Axi-symmetry Navier-Stokes and convective/diffusive equations were used.

Constants and values used:

Length of pipe L	19.4 cm
Radius of pipe R	0.15 cm
Viscosity μ	$3.85*10^{-3}$ Pa.s
Frequency f	60 Hz
Density ρ	$1.2g/cm^3$
Vortex viscosity ζ	0.3*m
Spin Viscosity η'	$8.67*10^{-9}$
Magnetic susceptibility χo	.0479 for H = 158 Oe
Time constant τ	$2.4x10^{-5}$
Magnetic field H	158 Oe
Flow rate Q	320 ml/min
Re	351

Boundaries Conditions:
(Navier-stokes)



Straight out

slip/
symmetr

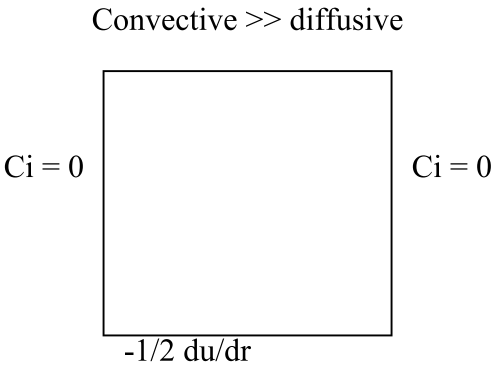
no-slip

fully develop flow

subdomain settings:

r = 1
n = 1 / Re
Fr = 0
Fz = 2 z/m(cz+c./r)

Convective/diffusive boundary settings



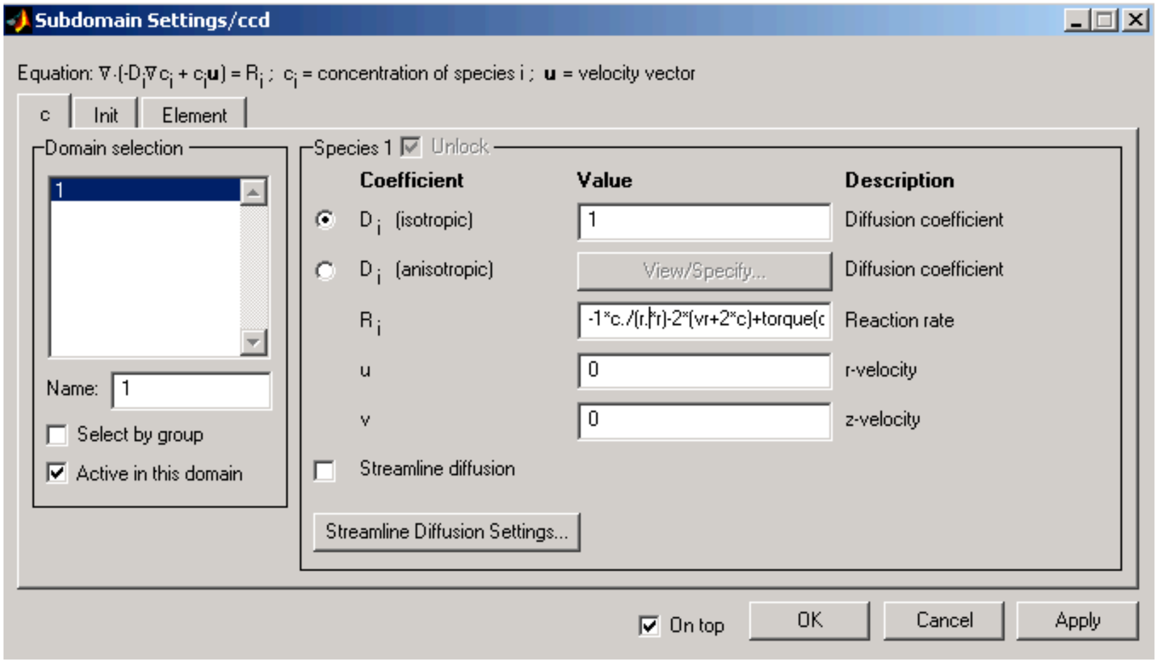
Convective >> diffusive

Ci = 0

Ci = 0

-1/2 du/dr

subdomain settings for convective/diffusive equation:



Subdomain Settings/ccd

Equation: $\nabla\cdot(-D_i\nabla c_i + c_i\mathbf{u}) = R_i$; c_i = concentration of species i ; $\mathbf{u}$ = velocity vector

Domain selection: 1

Name: 1

☐ Select by group

☑ Active in this domain

Species 1 ☑ Unlock

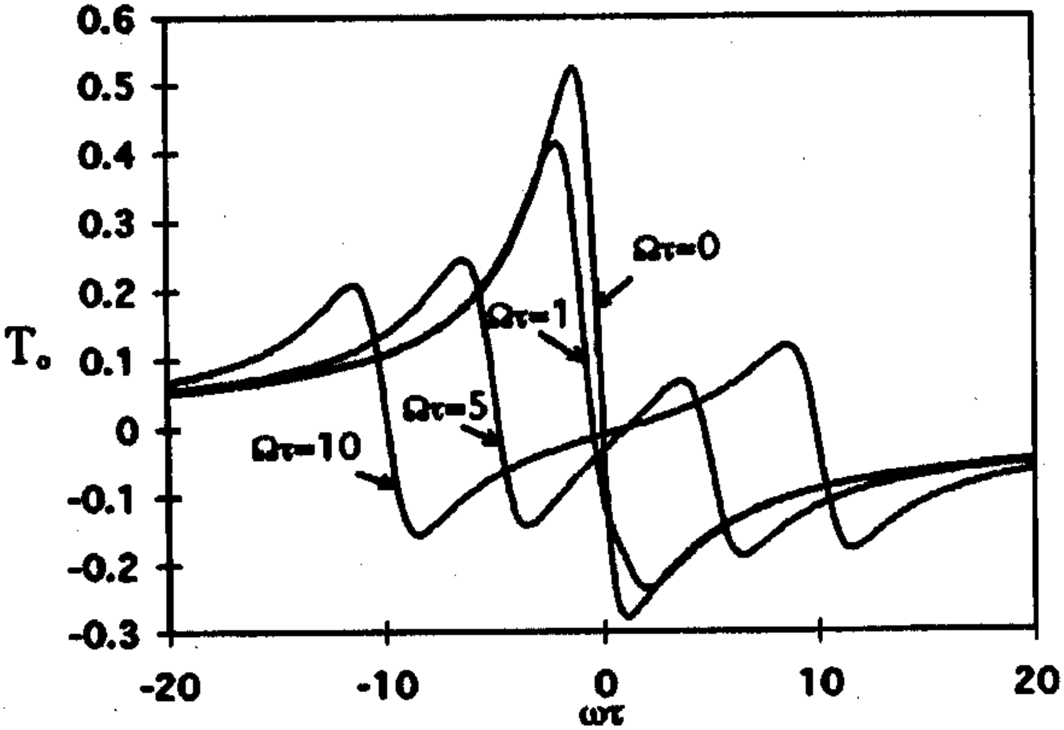
	Coefficient	Value	Description
●	D_i (isotropic)	1	Diffusion coefficient
○	D_i (anisotropic)	View/Specify...	Diffusion coefficient
	R_i	-1*c./(r.*r)-2*(vr+2*c)+torque(c	Reaction rate
	u	0	r-velocity
	v	0	z-velocity

☐ Streamline diffusion

☑ On top

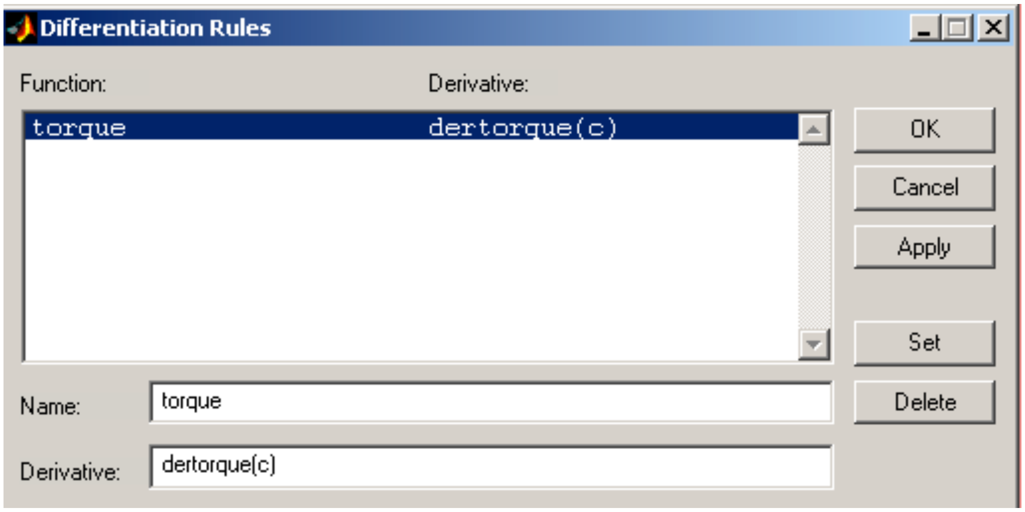
Due to difficulties encountered, the goal was unable to be fulfilled. However, a valuable technique was learned instead. A Matlab m-file was inserted into Femlab to take into account the torque term.

The torque m-file was first generated in Matlab. It was very carefully checked by putting in a matrix from -20 to 20 with increments of 0.25. Every constant was checked thoroughly by hand, and compared with the value printed out in Matlab. The last check was to plot spin versus torque, and a graph similar to the one shown below was created.



The next step was to insert the torque m-file into femlab. A series of difficulties were encountered. The first error message encountered was “undefined function torque”. Make sure to set Matlab search path to where the torque m-file was saved.

The next problem encountered was, “no differentiation rules set”
The solution to this problem was to a make another m-file that is the derivative of the torque m-file. Hence, dertorque.m was created. Again, numerical calculations checks were made. The dertorque file was inserted under “options – differentiation rules”.



Function:	Derivative:
torque	dertorque(c)

Name: torque

Derivative: dertorque(c)

Whilst there is no more calculation problems encountered, it still wasn't displaying the results expected. The torque term did not make any changes in the solution. It was later found out that it was not large enough to cause an effect. Therefore, the proposed recommendation is to use a larger torque term.

Appendix:

(A) Torque.m

```
%Create an m-file for torque
%paramters OMEGATAU (frequency of electric field)
%Xo (magnetic equation of state)
%Variable spin (wt)
%save as torque.m

function torque = torque(spin)
global OMEGATAU Xo

T = .024
OMEGATAU = .00905
Xo = .0479

aa = (spin*T).*(spin*T) - OMEGATAU*OMEGATAU + 1 + Xo
bb = OMEGATAU*(2 + Xo)
cc = (spin*T) + 1
dd = OMEGATAU
ff = OMEGATAU
ee = (1+Xo) - (spin*T)

torque =0.5*Xo*((aa.*(ee-cc))-Xo*(cc.*ee + dd*ff))./(((aa.^2)+ bb^2))
```

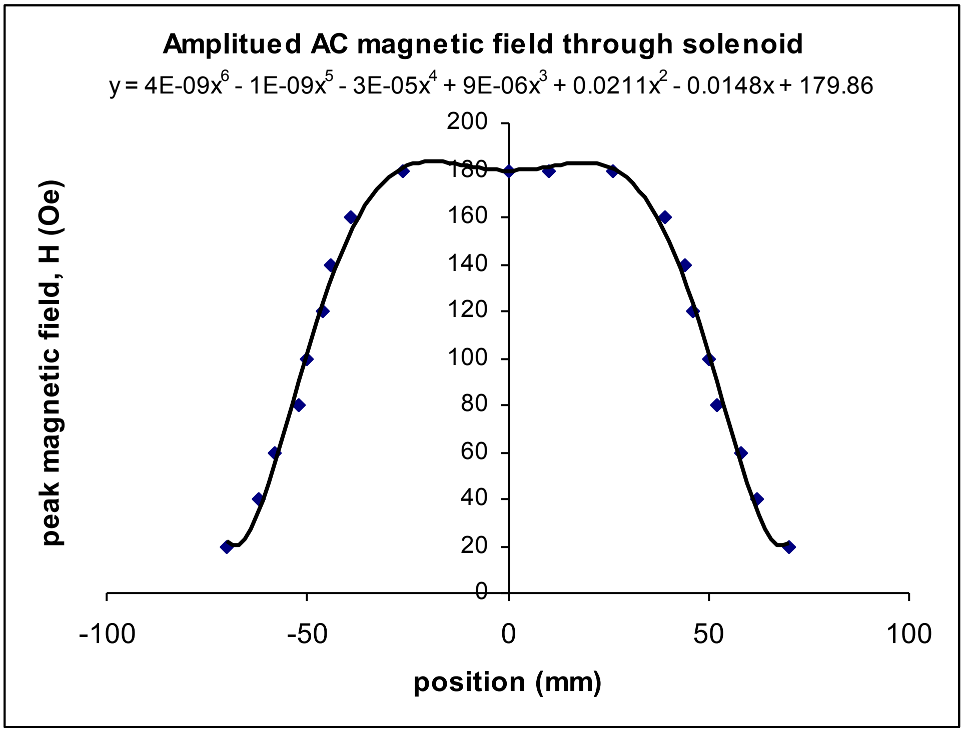
(B) dertorque.m

```
function dertorque = dertorque(spin)
global OMEGATAU Xo T

T = .024
OMEGATAU = .00905
Xo = .0479

aa = (spin*T).*(spin*T) - OMEGATAU*OMEGATAU + 1 + Xo
bb = OMEGATAU*(2 + Xo)
cc = (spin*T) + 1
dd = OMEGATAU
ff = OMEGATAU
ee = (1+Xo) - (spin*T)

a2 = 2*spin*T*T
b2 = 0
c2 = T
d2 = 0
e2 = -T
f2 = 0

bottom = ((2*a2.*aa).*(2*a2.*aa)) + ((2*b2.*bb).*(2*b2.*bb))

dertorque = .5*Xo*( ((aa.*aa + bb.*bb).*((a2*(e2-c2)) + (aa.*(e2-c2)) - Xo*( (c2*ee +
e2*cc) + (d2*ff + f2*dd) ))./ bottom)
```

(C) AC magnetic function

Position (mm)	H (Oe)
20	-70
40	-62
60	-58
80	-52
100	-50
120	-46
140	-44
160	-39
180	-26
180	0
180	10
180	26
160	39
140	44
120	46
100	50
80	52
60	58
40	62
20	70

This is the function planned to be used for the non-steady magnetic field.



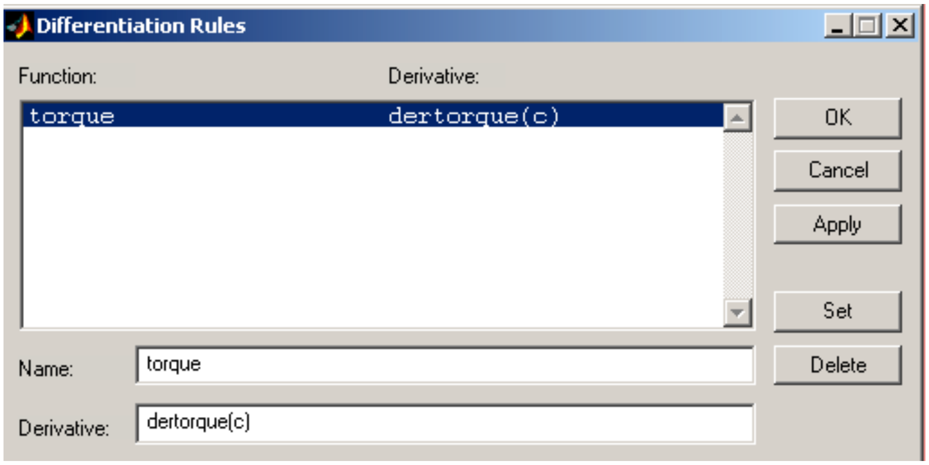
(D) Using a Matlab m-file in femlab

1. Create the matlab m-file
2. Creat another m-file that gives the derivative of the first function.
3. Insert the m-file into the required field
4. Insert the derivative m-file under

   “Options…… differentiation rules.

   Name: <file name of function>

   Derivative: <file name of derivative of function> (parameter)



Function:	Derivative:
torque	dertorque(c)

Name: torque

Derivative: dertorque(c)