OOMMF

2D Solver

Advanced

Details

Solver Module Features

- 3D spins on a square 2D grid
- LLG ODE solver
- Demag calculated via FFT
- Anisotropy, applied field, and initial magnetization ptwise selectable
- Shaped elements

Micromagnetic Equations

Landau-Lifshitz-Gilbert:

$$\frac{d\mathbf{M}}{dt} = \frac{-\omega}{1+\lambda^2}\mathbf{M}\times\mathbf{H}_{\text{eff}} - \frac{\lambda\,\omega}{(1+\lambda^2)M_{\text{s}}}\mathbf{M}\times(\mathbf{M}\times\mathbf{H}_{\text{eff}})$$

$$\mathbf{H}_{\text{eff}} = -\frac{1}{\mu_0}\frac{\partial E_{\text{density}}}{\partial \mathbf{M}}$$

Energies:

$$E_{\text{exchange}} = \frac{A}{M_s^2}\left(|\nabla M_x|^2 + |\nabla M_y|^2 + |\nabla M_z|^2\right)$$

$$E_{\text{anis}} = \frac{K_1}{M_s^4}(M_x^2M_y^2 + M_y^2M_z^2 + M_z^2M_x^2)$$

$$E_{\text{demag}} = \frac{\mu_0}{8\pi}\mathbf{M}(r)\cdot\left[\int_V \nabla\cdot\mathbf{M}(\mathbf{r}')\frac{\mathbf{r}-\mathbf{r}'}{|\mathbf{r}-\mathbf{r}'|^3}\,d^3r' - \int_S \hat{\mathbf{n}}\cdot\mathbf{M}(\mathbf{r}')\frac{\mathbf{r}-\mathbf{r}'}{|\mathbf{r}-\mathbf{r}'|^3}\,d^2r'\right]$$

$$E_{\text{Zeeman}} = -\mu_0\mathbf{M}\cdot\mathbf{H}_{\text{ext}}$$

Sample MIF 1.1 File

```
# MIF 1.1
# material name:Custom
ms:1000e3
k1:0
anisotropy type:uniaxial
anisotropy dir1:0 0 1
demag type:constmag

part height:5e-6
part width:1e-6
part thickness:100e-9
part shape:rectangle
cell size:20e-9

init mag:uniform 85.0 63.8
randomizer seed:1

default control point spec: -torque 1e-5
field range: .1 .1 .1 0.01 0.01 0.01 9
field range: 0.01 0.01 0.01 0. 0. 0. 1 -torque 1e-6
field range: 0 0 0 -.03 -.03 -.03 6
field range: -.03 -.03 -.03 -.045 -.045 -.045 30
field range: -.045 -.045 -.045 -.05 -.05 -.05 1
field range: -.05 -.05 -.05 -.1 -.1 -.1 5

a:6.981317e-10
base output filename: prob2dlex30.00
user comment: muMag Problem 2, d/lex=30.0
```

Sample MIF edits

- **Do Precess: 1**
- **gyratio: 2.21e5** in m/(A.s)
- **Default Control Point Spec: -torque 1e-5**
  and/or -time, -iteration
- **Field Type: OneFile *filename mult***

MIF 1.1 Files: Multiple Applied Fields

```
Part Width:     0.25e-6    # Nominal part width in m
Part Height:    1.0e-6     # Nominal part height in m
Part Thickness: 1e-9       # Part thickness in m.
Cell Size:      8.1e-9     # Cell size in m.

Field Type: Multi 4 \
 7 Ribbon 1 0 1.0e-6 0.25e-6 1.0e-6 1e-9 \
 7 Ribbon 1 0 0     0.25e-6 0     1e-9 \
 9 Tie 100 0 0 0.12e-6 0.5e-6 0.13e-6 0.5e-6 8.1e-9 \
 1 Uniform
# The above positions ribbons of positive charge along the upper
# and lower edges with strength Ms, applies a large (100 Ms) field
# to the center cell, and also applies a uniform field across the
# sample stepped from (-.05,-.01,0.) to (.05,.01,0.) (Tesla), and
# back, in approximately 0.001 T steps.
```

MIF 1.1 Files: Dynamics

```
# MIF 1.1
# material name:Custom
ms:8e5
a:13E-12
k1:0
damp coef:0.02
#damp coef:0.5
anisotropy type:uniaxial
anisotropy init:constant
anisotropy dir1:1 0 0
anisotropy dir2:0 1 0
demag type:constmag
part height:500E-9
part width:125E-9
part thickness:3e-9
part shape:rectangle
#cell size:3.125e-9
#cell size:2.5e-9
cell size: 5e-9
init mag:avffile app/mmpe/examples/prob4-start.omf
base output filename:prob4
magnetization output format:binary 4
total field output format:binary 4
data table output format:%.15g
randomizer seed:0
randomizer seed:0
max time step: 0.2e-12
field range: 0 0 0 0 0 0 0 -torque 1e-9
field range: 0.0063 -0.0355 0 0.0063 -0.0355 0 5000 -time 1e-12
```

Sample Batch Launch Commands

- Process one MIF file:
  tclsh oommf.tcl batchsolve taskA.mif
- Process a sequence of MIF files:
  tclsh oommf.tcl batchmaster simpletask.tcl

Sample Batch Task File

```
# FILE: prob2task.tcl
#
# This is a sample batch task file.  Usage example:
#
#        oommf.tcl -fg batchmaster -tk 0 prob2task.tcl
#
# LOCAL MODIFICATIONS: It is recommended you copy this sample file and
#   make modifications to the copy (which is then specified on the
#   batchmaster.tcl command line in place of prob2task.tcl).  You will
#   want to change
#
# $TaskInfo SetSlaveInitScript --- This script is sent to each slave
#                when it is first brought up.  Do any one-time
#                initialization here, but also define the procs
#                SolverTaskInit, SolverTaskCleanup and optionally
#                BatchTaskRelaxCallback.  Use SolverTaskInit to
#                perform task-specific initialization.
# $TaskInfo AppendTask --- List of tasks.  Each entry is a pair
#                consisting of a task label (id), and a Tcl script to
#                eval.

# Task script configuration
set BASEMIF prob2     ;# Base mif name
#set CELLSIZE 20       ;# Simulation cellsize in nm
#set dlex_list { 30 25 20 15 10 5 }
#set CELLSIZE 40       ;# Simulation cellsize in nm
#set dlex_list { 15 10 5 }
set CELLSIZE 250       ;# Simulation cellsize in nm
set dlex_list { 0.25 0.125 }


# Slave initialization script (with batchsolve.tcl proc redefinitions)
set init_script {
    # Initialize solver. This is run at global scope
    set basename __BASEMIF__      ;# Base mif filename (global)
    set cellsize __CELLSIZE__     ;# Simulation discretization size
    mms_mif New mif
    $mif Read [FindFile ${basename}.mif]
    # Redefine TaskInit and TaskCleanup proc's
    proc SolverTaskInit { args } {
global mif outtextfile basename cellsize
```

```
set dlex [lindex $args 0]
set outbasename "$basename-dlex$dlex-cellsize$cellsize"
# Convert d/lex to A
set tmp [expr [$mif GetPartWidth]*[$mif GetMs]/$dlex]
set A [expr $tmp*$tmp*6.2831853e-7]
$mif SetA $A
$mif SetCellSize [expr $cellsize*1e-9]  ;# Convert to meters
$mif SetOutBaseName $outbasename
$mif SetUserComment "muMag Problem 2, d/lex=$dlex,\
cellsize=$cellsize nm"
set outtextfile [open "$outbasename.odt" "a+"]
puts $outtextfile [GetTextData header \
  "mmSolve run on $basename.mif, for d/lex=$dlex (A=[$mif GetA])"]
flush $outtextfile
    }
    proc SolverTaskCleanup { args } {
        global outtextfile
        close $outtextfile
    }
}
# Substitute $BASEMIF in for __BASEMIF__ in above script
regsub -all -- __BASEMIF__ $init_script $BASEMIF init_script
# Substitute $CELLSIZE in for __CELLSIZE__ in above script
regsub -all -- __CELLSIZE__ $init_script $CELLSIZE init_script

# Set init script into TaskInfo
$TaskInfo SetSlaveInitScript $init_script

# Create task list
foreach dlex $dlex_list {
    $TaskInfo AppendTask "d/lex=$dlex" "BatchTaskRun $dlex"
}
```