OOMMF Tutorial
Part II: OOMMF Basics

Michael J. Donahue

Applied and Computational Mathematics Division
National Institute of Standards and Technology
Gaithersburg, Maryland

26-May-2020

Special Thanks to

Online Spintronics Seminar

- Professor Xin Fan (Univ. Denver)
- Professor Kirill Belashchenko (Univ. Nebraska-Lincoln)

nanoHUB

- Tanya Faltens (Purdue University)

IEEE Magnetics Society

DISCLAIMER: The mention of specific products, trademarks, or brand names is for purposes of identification only. Such mention is not to be interpreted in any way as an endorsement or certification of such products or brands by the National Institute of Standards and Technology. All trademarks mentioned herein belong to their respective owners.

OOMMF Tutorial
Part II: OOMMF Basics

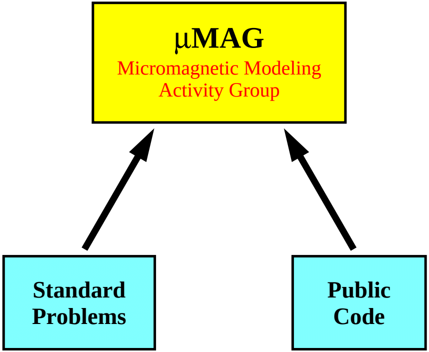
Session schedule

- Thur, 21-May-2020: Intro to Micromagnetics
- Tues, 26-May-2020: OOMMF Basics
- Tues, 2-June-2020: Pitfalls, writing an extension, batch processing
- Tues, 9-June-2020: Data analysis, pics, movies, dispersion curves, . . .

All sessions start at 12:00 noon EDT.

OOMMF Tutorial
Part II: OOMMF Basics

μMAG

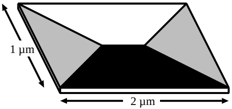


Center for Theoretical and Computational Materials Science
http://www.ctcms.nist.gov/

See also the mailing list and archives!

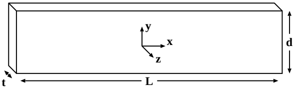
μMAG standard problems

Problem 1: Hysteresis



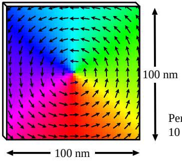
Permalloy rectangle, 20 nm thick

Problem 2: Hysteresis



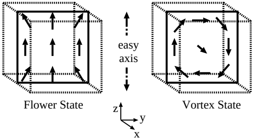
L/d = 5.0 **t/d=0.1**

Problem 5: CIP Spin Torque

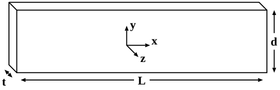


Permalloy square, 10 nm thick

Problem 3: Energy Minimization



Problem 4: Dynamics



L=500 nm **d=125 nm** **t=3 nm**

Portable, extensible, public domain programs & tools for micromagnetics

https://math.nist.gov/oommf

Primary developers: Don Porter, Mike Donahue (NIST)

- Finite difference code
- Graphical User Interface
- Windows, Unix, macOS
- Binaries and source code
- Tcl/Tk and C++ based modular architecture
- 250 page user's manual
- Available at nanoHUB

- Extensible architecture: numerous third party extensions

Portable, extensible, public domain programs & tools for micromagnetics

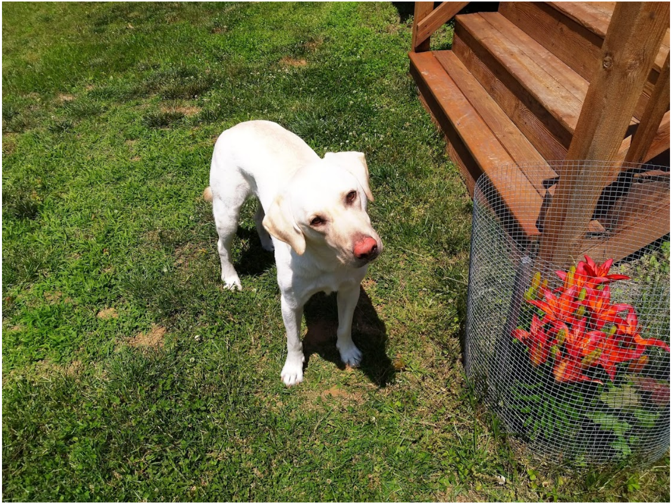
https://math.nist.gov/oommf

Primary developers: Don Porter, Mike Donahue (NIST)

- Finite difference code
- Graphical User Interface
- Windows, Unix, macOS
- Binaries and source code
- Tcl/Tk and C++ based modular architecture
- 250 page user's manual
- Available at nanoHUB

- Extensible architecture: numerous third party extensions
- If you use OOMMF, please cite!

Please cite!

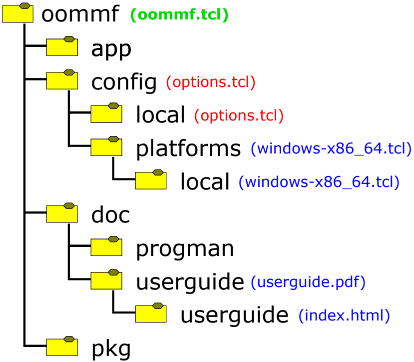


Do it for the dog!

OOMMF file layout

- oommf **(oommf.tcl)**
  - app
  - config
  - doc
  - pkg

OOMMF file layout



OOMMF file layout

- oommf **(oommf.tcl)**
  - app
    - mmarchive
    - mmdatatable
    - mmdisp
    - mmgraph
    - mmlaunch
    - mmpe
    - oxs
      - base
      - contrib
      - examples
      - ext
      - local
      - windows-x86_64
  - config
  - doc
  - pkg

OOMMF Tutorial
Part II: OOMMF Basics

OOMMF

Widget demo

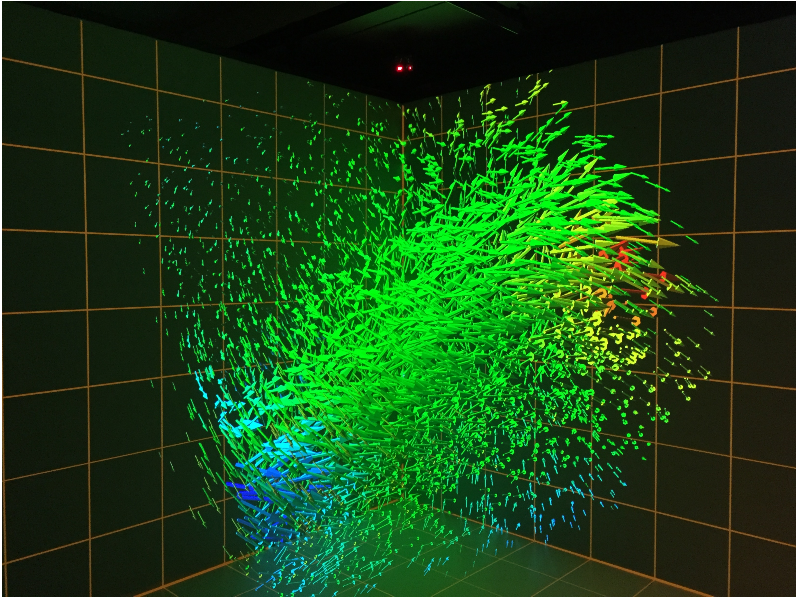
MIF files

Homework

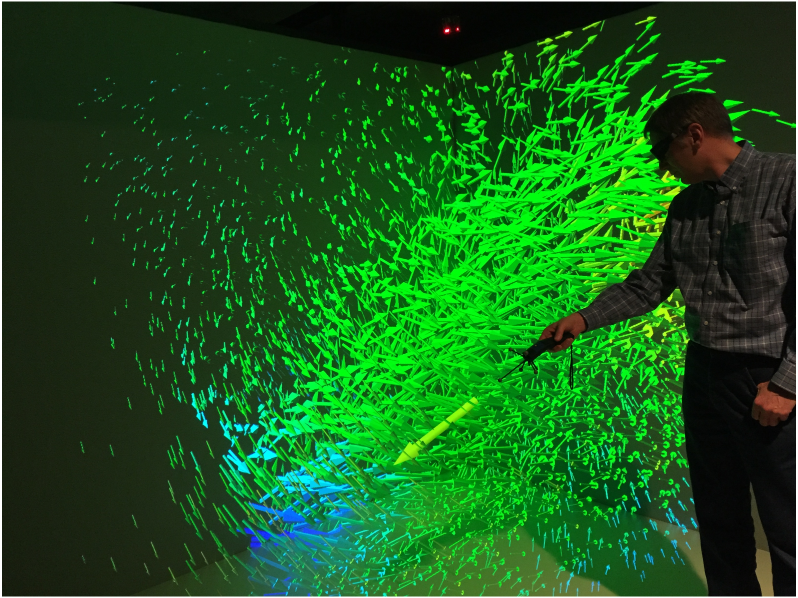
Widget overview

- mmLaunch
- Oxsii
- mmDisp
- mmDataTable
- mmGraph
- mmArchive
- mmProbEd

3D visualization



3D visualization



OOMMF Tutorial
Part II: OOMMF Basics

MIF file (Standard Problem 4)

```
# MIF 2.1
set pi [expr {4*atan(1.0)}]
set mu0 [expr {4*$pi*1e-7}]
set Hx  -35.5  ;# Applied field in mT
set Hy   -6.3  ; set Hz    0.0

Parameter xcellsize 2.5e-9
Parameter ycellsize 2.5e-9
Parameter zcellsize   3e-9

Specify Oxs_BoxAtlas:atlas {
  xrange {0 500e-9}
  yrange {0 125e-9}
  zrange {0   3e-9}
}
Specify Oxs_RectangularMesh:mesh [subst {
  cellsize {$xcellsize $ycellsize $zcellsize}
  atlas Oxs_BoxAtlas:atlas
}]

Specify Oxs_UniformExchange {
  A  13E-12
}
Specify Oxs_Demag {}
Specify Oxs_FixedZeeman [subst {
    multiplier [expr {0.001/$mu0}]
    field {$Hx $Hy $Hz}
}]

Specify Oxs_FileVectorField:m_init {
   atlas :atlas
   norm  1.0
   file  sp4-start.omf
}

Specify Oxs_RungeKuttaEvolve:evolver {}
Specify Oxs_TimeDriver {
   evolver :evolver
   mesh :mesh
   stopping_dm_dt 0.01
   Ms 8e5
   m0 :m_init
}
```

MIF file - annotated

MIF Header →

```
# MIF 2.1
```

Tcl set up ←

```
set pi [expr {4*atan(1.0)}]
set mu0 [expr {4*$pi*1e-7}]
set Hx  -35.5  ;# Applied field in mT
set Hy   -6.3  ; set Hz    0.0
```

Command line params →

```
Parameter xcellsize 2.5e-9
Parameter ycellsize 2.5e-9
Parameter zcellsize   3e-9
```

Atlas (≥ 1) ←

```
Specify Oxs_BoxAtlas:atlas {
  xrange {0 500e-9}
  yrange {0 125e-9}
  zrange {0   3e-9}
}
```

Mesh (1) →

```
Specify Oxs_RectangularMesh:mesh [subst {
  cellsize {$xcellsize $ycellsize $zcellsize}
  atlas Oxs_BoxAtlas:atlas
}]
```

Energy terms (as needed) ←

```
Specify Oxs_UniformExchange {
  A  13E-12
}
Specify Oxs_Demag {}
Specify Oxs_FixedZeeman [subst {
    multiplier [expr {0.001/$mu0}]
    field {$Hx $Hy $Hz}
}]
```

Vector field (as needed) →

```
Specify Oxs_FileVectorField:m_init {
   atlas :atlas
   norm  1.0
   file  sp4-start.omf
}
```

Evolver (1) ←

```
Specify Oxs_RungeKuttaEvolve:evolver {}
```

Driver (1) →

```
Specify Oxs_TimeDriver {
   evolver :evolver
   mesh :mesh
   stopping_dm_dt 0.01
   Ms 8e5
   m0 :m_init
}
```

< Optional output requests >

Tour of Oxs_Ext classes

- Atlases

  Oxs_BoxAtlas Oxs_ImageAtlas
  Oxs_ScriptAtlas Oxs_MultiAtlas

- Meshes

  Oxs_RectangularMesh Oxs_PeriodicRectangularMesh

- Scalar fields

  Oxs_UniformScalarField Oxs_AtlasScalarField
  Oxs_ScriptScalarField Oxs_RandomVectorField
  Oxs_VecMagScalarField Oxs_ImageScalarField

- Vector fields

  Oxs_UniformVectorField Oxs_AtlasVectorField
  Oxs_ScriptVectorField Oxs_RandomVectorField
  Oxs_MaskVectorField Oxs_FileVectorField

Tour of Oxs_Ext classes (cont.)

- Energies

Oxs_UniaxialAnisotropy	Oxs_Demag
Oxs_UZeeman	Oxs_StageZeeman
Oxs_UniformExchange	Oxs_Exchange6Ngbr
Oxs_TwoSurfaceExchange	Oxs_DMExchange6Ngbr

- Evolvers

Oxs_CGEvolve	Oxs_RungeKuttaEvolve
Anv_SpinTEvolve	

- Drivers

Oxs_MinDriver	Oxs_TimeDriver

Intro to Tcl


```
% set a 5
5
% set b 3
3
% set c [expr {$a+$b}]
8
% puts "The sum of $a and $b is $c"
The sum of 5 and 3 is 8
% puts {The sum of $a and $b is $c}
The sum of $a and $b is $c
% puts [subst {The sum of $a and $b is $c}]
The sum of 5 and 3 is 8
% incr b
4
% incr b 2
6
% incr b -3
3
% for {set a 1; set sum 1} {$a<=5} {incr a} {
      set sum [expr {$sum*$a}]
  }
% puts $sum
120
% if {$sum<100} {
      puts "sum=$sum is small"
  } else {
      puts "sum=$sum is big"
  }
sum=120 is big
% exit
```

Intro to Tcl

```
% set a 5
5
% set b 3
3
% set c [expr {$a+$b}]
8
% puts "The sum of $a and $b is $c"
The sum of 5 and 3 is 8
% puts {The sum of $a and $b is $c}
The sum of $a and $b is $c
% puts [subst {The sum of $a and $b is $c}]
The sum of 5 and 3 is 8
% incr b
4
% incr b 2
6
% incr b -3
3
% for {set a 1; set sum 1} {$a<=5} {incr a} {
      set sum [expr {$sum*$a}]
   }
% puts $sum
120
% if {$sum<100} {
      puts "sum=$sum is small"
   } else {
      puts "sum=$sum is big"
   }
sum=120 is big
% exit
```

command
argument 1
argument 2
white space
literal value
variable
execute enclosed command
expr command
variable substitution
command
quoted element
braced element
command
incr command (integers only)
for command
cmd arg
expr arg
cmd arg
cmd arg
if command
expr arg
cmd arg
cmd arg

Intro to Tcl

```
% list answers {}
answers {}
% proc factorial { n } {
  # Factorial routine; appends results to answers
  global answers
  set prod 1
  while {$n>1} {
    set prod [expr {$prod*$n}]
    incr n -1
  }
  lappend answers $prod
  return $prod
}
% factorial 7
5040
% factorial 3
6
% factorial 10
3628800
% puts $answers
5040 6 3628800
```

Tcl references

https://www.tcl-lang.org/

https://wiki.tcl-lang.org/page/Online+Tcl+and+Tk+Tutorials

https://wiki.tcl-lang.org/page/Dodekalogue

OOMMF Tutorial
Part II: OOMMF Basics

Homework: Problem 1, Part A

Write a MIF file for this problem:

Part dimensions: 500 nm x 200 nm x 0.6 nm
Ms=1.1e6 A/m, A=1.6e-11 J/m
K1=5.1e5 J/m^3 along the (0,0,1) axis
DMI: D=3.5e-3 J/m^2, Free boundaries

Use the Oxs_DMExchange6Ngbr extension to model the DMI.
https://www.lps.u-psud.fr/spip.php?article2252

Initial magnetization configuration: Ignoring z-coords, let P be the point (50 nm, 50 nm) relative to the lower left hand corner of the simulation. Set m=(0,0,1) for all points closer to P than 16 nm. Set m=(0,0,-1) for all points farther from P than 23 nm. For points in-between, set m to point towards P. Write a Tcl proc to use with Oxs_ScriptVectorField to set up this initial configuration.

Homework: Problem 1, Part A (cont.)

Use Oxs_CGEvolve to relax the initial state towards equilibrium. Try different cell sizes in the range 1 nm to 4 nm. The magnetization should relax into a skyrmion. If the skyrmion forms but wanders away from the initial location, introduce a small region with larger K1 near P to pin the skyrmion. See how small K1 needs to be to hold the skyrmion in place.

In Part B we will introduce a spin current to move the skyrmion.