Standard Problems and Public Code for Micromagnetics

Michael J. Donahue, Donald G. Porter

Information Technology Laboratory, NIST

Robert D. McMichael

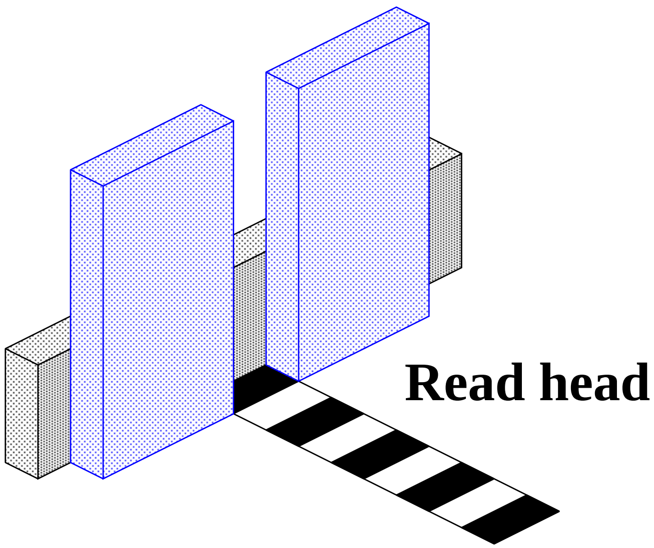
Materials Science & Engineering Laboratory, NIST

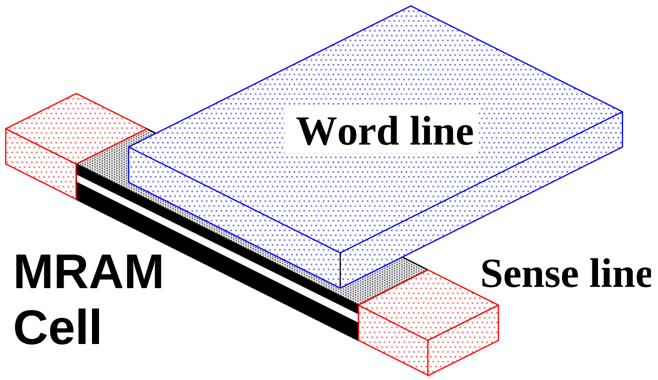
Overview

- Micromagnetics background
- μMAG
  - Standard problems
  - Public code

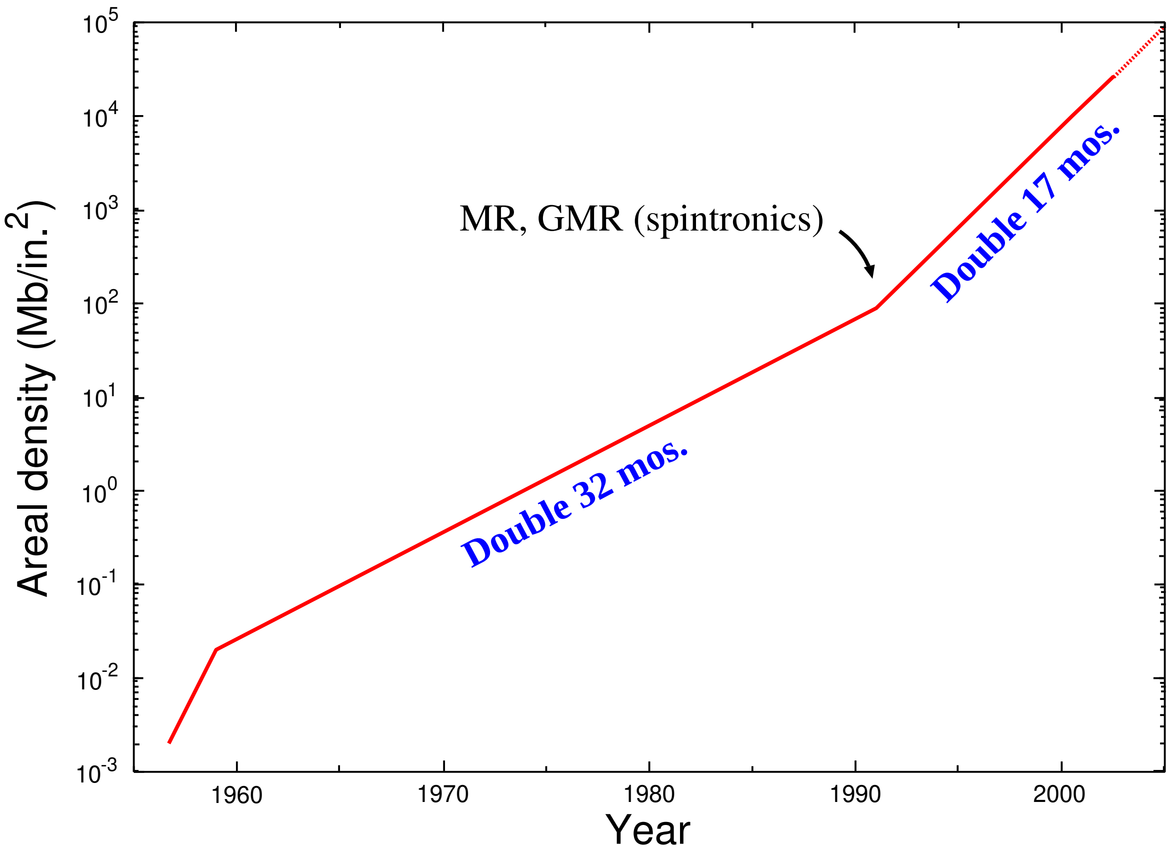
Why Computational Micromagnetics?

- Disk Drives
- Sensors
- Nonvolatile Memory
- Spintronics

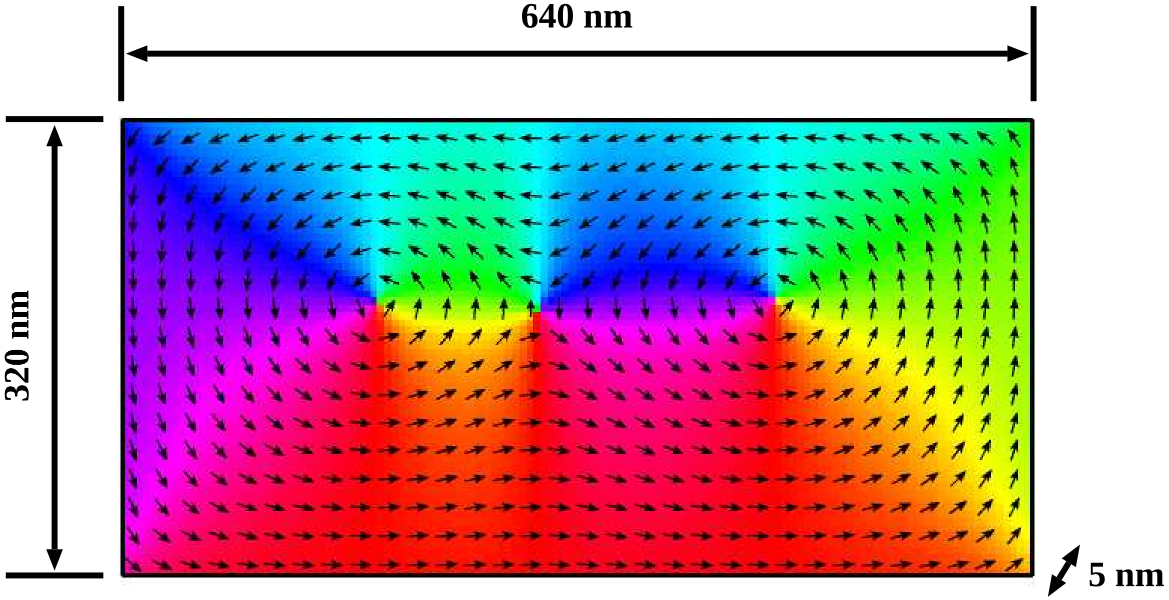




Magnetic Disk Storage



Micromagnetics. . .



. . . is the study, modeling and simulation of magnetic materials and their behavior at the nanometer scale.

Brown's Equations

Energies:

$$
\begin{aligned}
E_{\text{exchange}} &= \frac{A}{M_s^2}\left(|\nabla M_x|^2 + |\nabla M_y|^2 + |\nabla M_z|^2\right) \\
E_{\text{anis}} &= \frac{K_1}{M_s^2}(\mathbf{M}\cdot\mathbf{u})^2 \\
E_{\text{demag}} &= \frac{\mu_0}{8\pi}\mathbf{M}(r)\cdot\left[\int_V \nabla\cdot\mathbf{M}(\mathbf{r}')\frac{\mathbf{r}-\mathbf{r}'}{|\mathbf{r}-\mathbf{r}'|^3}\,d^3r'\right. \\
&\qquad \left. - \int_S \hat{\mathbf{n}}\cdot\mathbf{M}(\mathbf{r}')\frac{\mathbf{r}-\mathbf{r}'}{|\mathbf{r}-\mathbf{r}'|^3}\,d^2r'\right] \\
E_{\text{Zeeman}} &= -\mu_0\mathbf{M}\cdot\mathbf{H}_{\text{ext}}
\end{aligned}
$$

Magnetization Dynamics

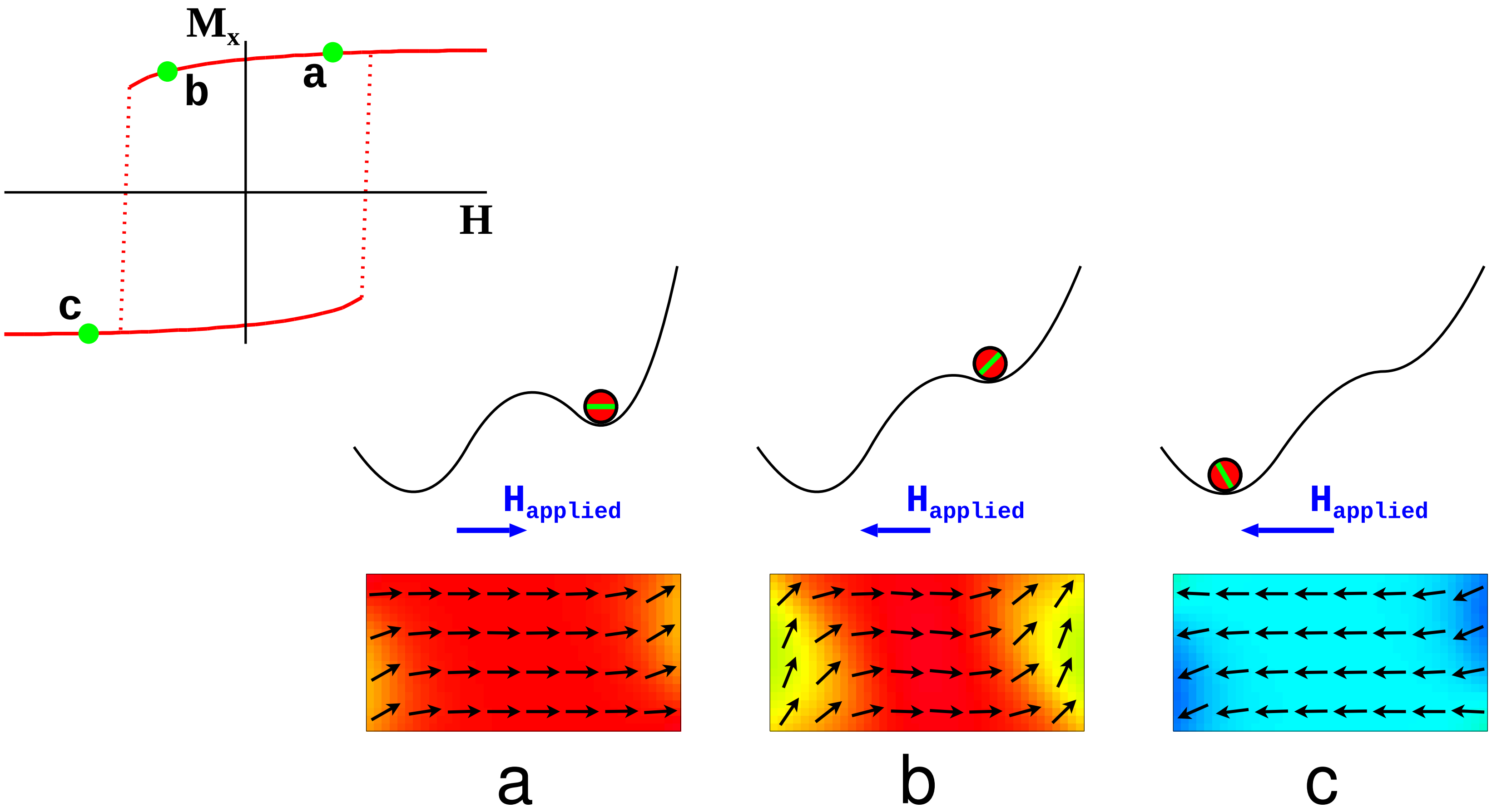
Landau-Lifshitz-Gilbert:

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

where

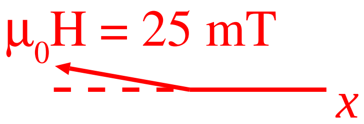
$$\begin{aligned} \mathbf{H}_{\text{eff}} &= -\frac{1}{\mu_0}\frac{\partial E}{\partial \mathbf{M}} \\ \omega &= \text{gyromagnetic ratio} \\ \lambda &= \text{damping coefficient} \end{aligned}$$

Quasi-Static Micromagnetics

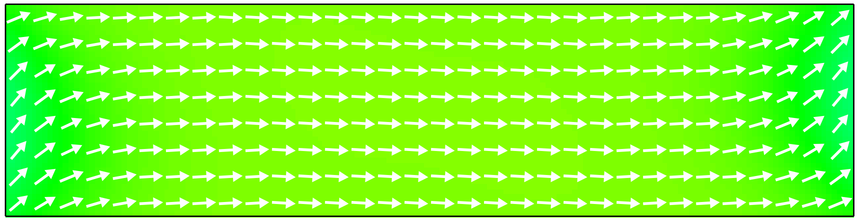


Magnetization Dynamics

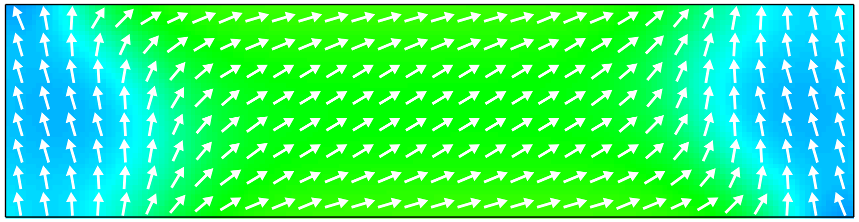
Time



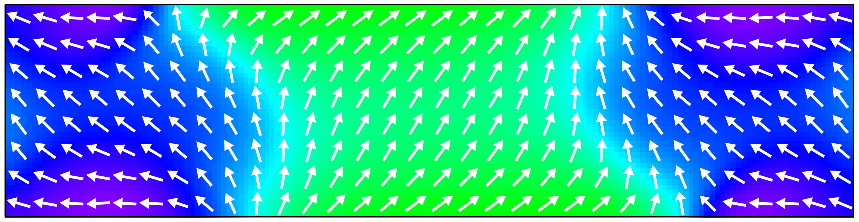
0 ps



100 ps

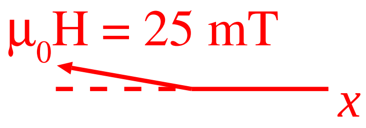


150 ps

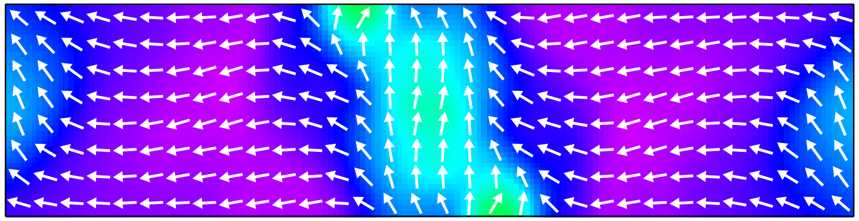


Magnetization Dynamics

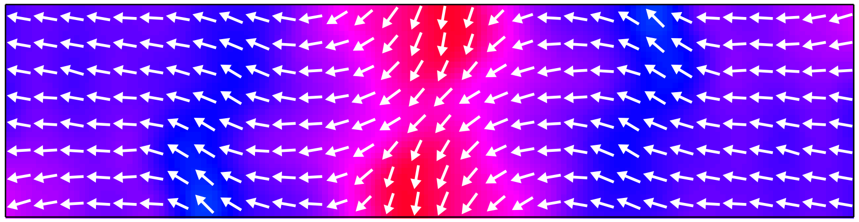
Time



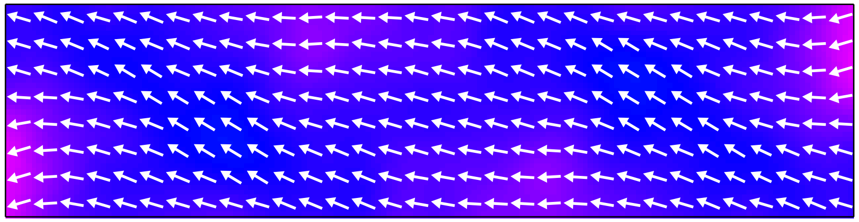
350 ps



450 ps



750 ps



Micromagnetics, ca 1995

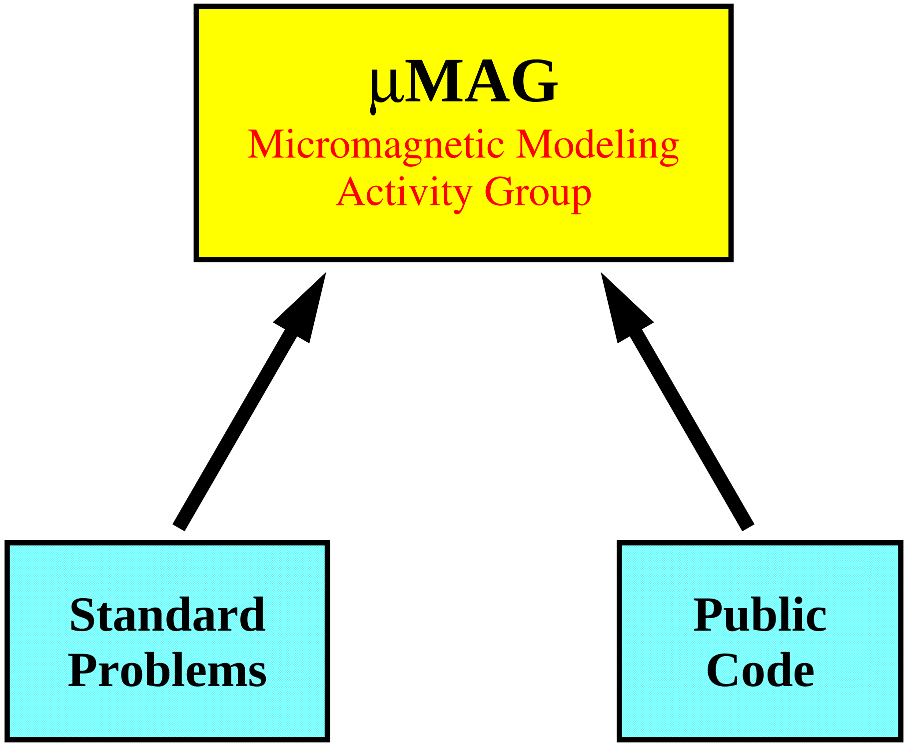
- No public or commercial codes

  $\Longrightarrow$ Reinvent wheel, straw men

- No two groups working on same problems
- Problems and methods poorly specified

  $\Longrightarrow$ Replication of results difficult

$\star$ Hard to judge quality of codes and techniques



Center for Theoretical and Computational Materials Science

http://www.ctcms.nist.gov/

μMAG Timeline

- Jan-1995 μMAG Steering Committee, Gaithersburg
- Apr-1995 First μMAG Workshop, San Antonio
- Nov-1995 μMAG Workshop, Philadelphia
- May-1996 Standard Problem No. 1
- Nov-1996 μMAG Workshop, Atlanta
- Mar-1997 Initial public release of mmDisp
- Jan-1998 μMAG workshop, San Francisco
- Jan-1998 OOMMF v1.0a0
- Feb-1998 Standard Problem No. 2
- Mar-1998 Standard Problem No. 3
- Aug-1998 Micromagnetics workshop, Boulder
- Oct-1998 OOMMF v1.0b0

μMAG Timeline

- Nov-1998 Standard problem session, MMM/Miami
- Apr-1999 OOMMF v1.1a0
- Feb-2000 Standard Problem No. 4
- Apr-2000 OOMMF v1.1b0
- Aug-2000 OOMMF Workshop, Gaithersburg
- Nov-2000 OOMMF v1.2a0
- Jan-2001 OOMMF v1.2a1
- May-2001 OOMMF v1.2a2
- Oct-2001 OOMMF v1.1b1
- Oct-2002 OOMMF v1.2a3
- Jan-2004 μMAG Workshop, Anaheim
- Jan-2004 OOMMF v1.1b2

Standard Problems

μMAG

Micromagnetic Modeling Activity Group

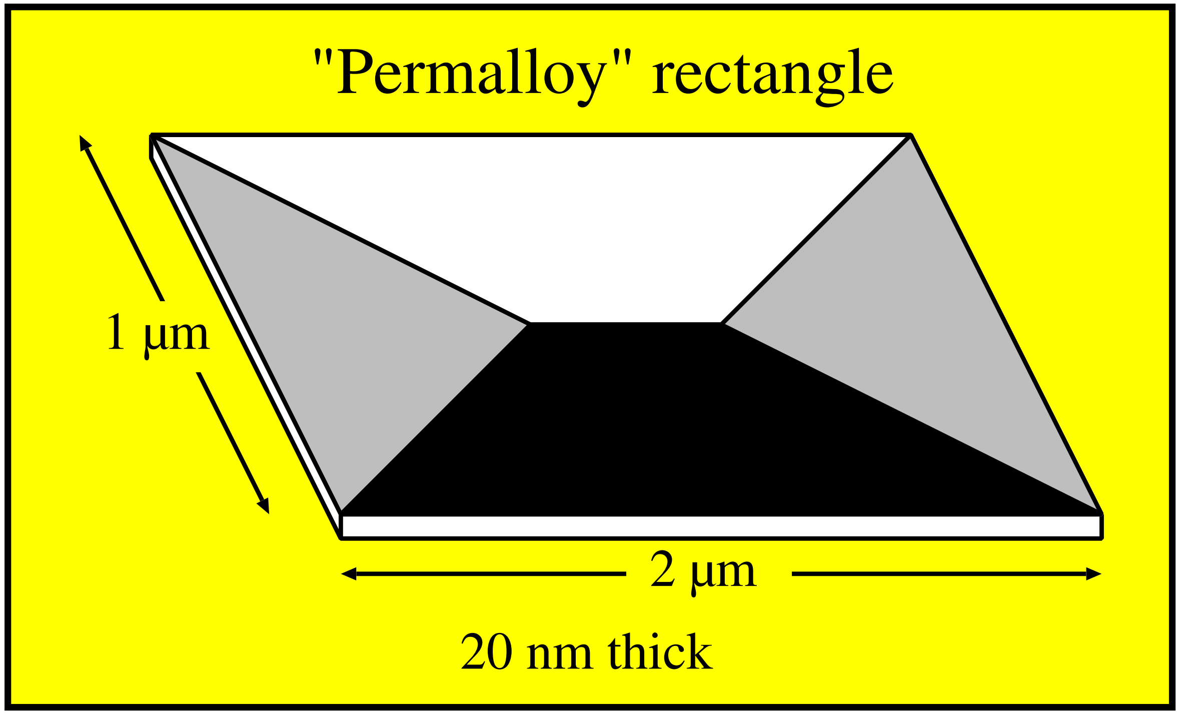
Four Standard Problems for micromagnetics

http://www.ctcms.nist.gov/~rdm/mumag.html

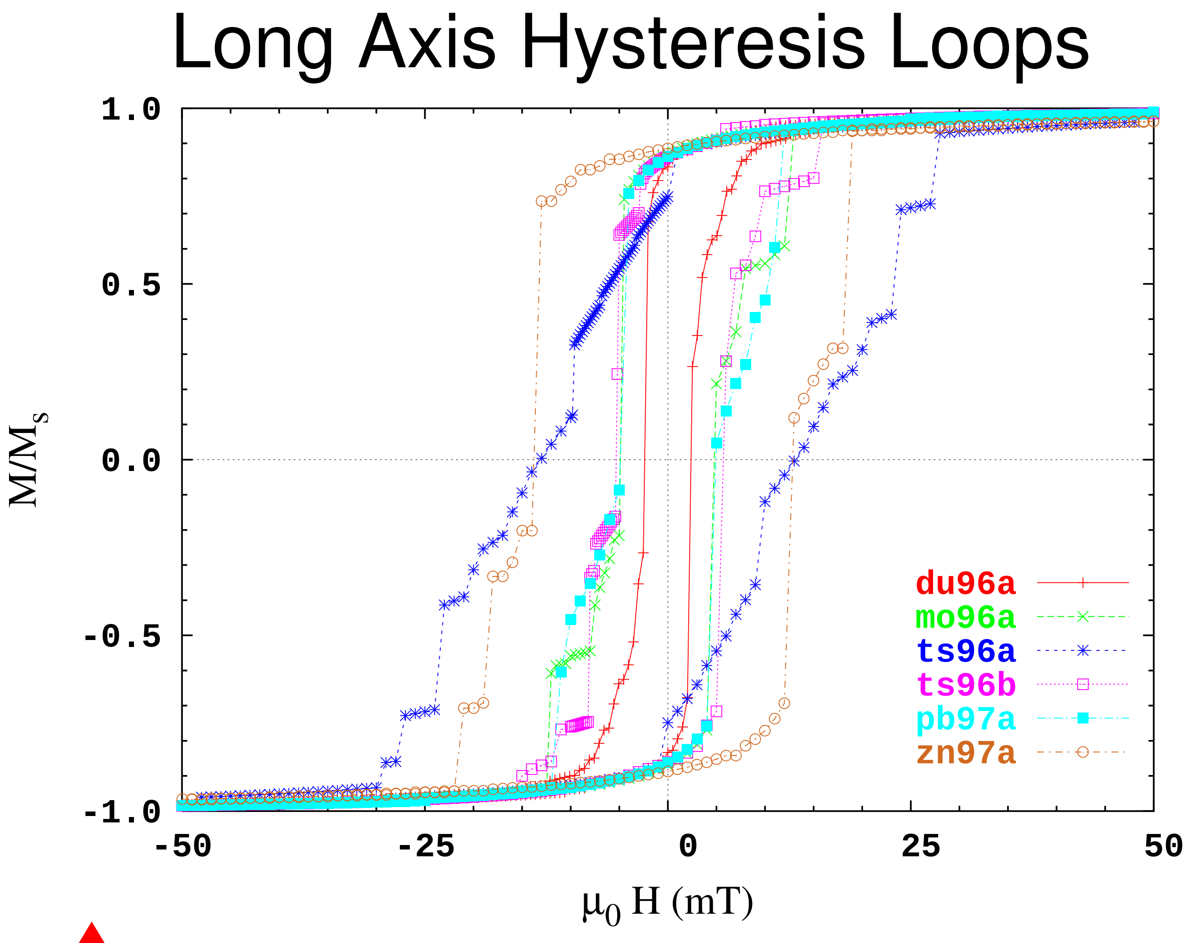
Check computed outputs against contributed solutions:

- Verify algorithms
- Compare methods
- Optimize parameters

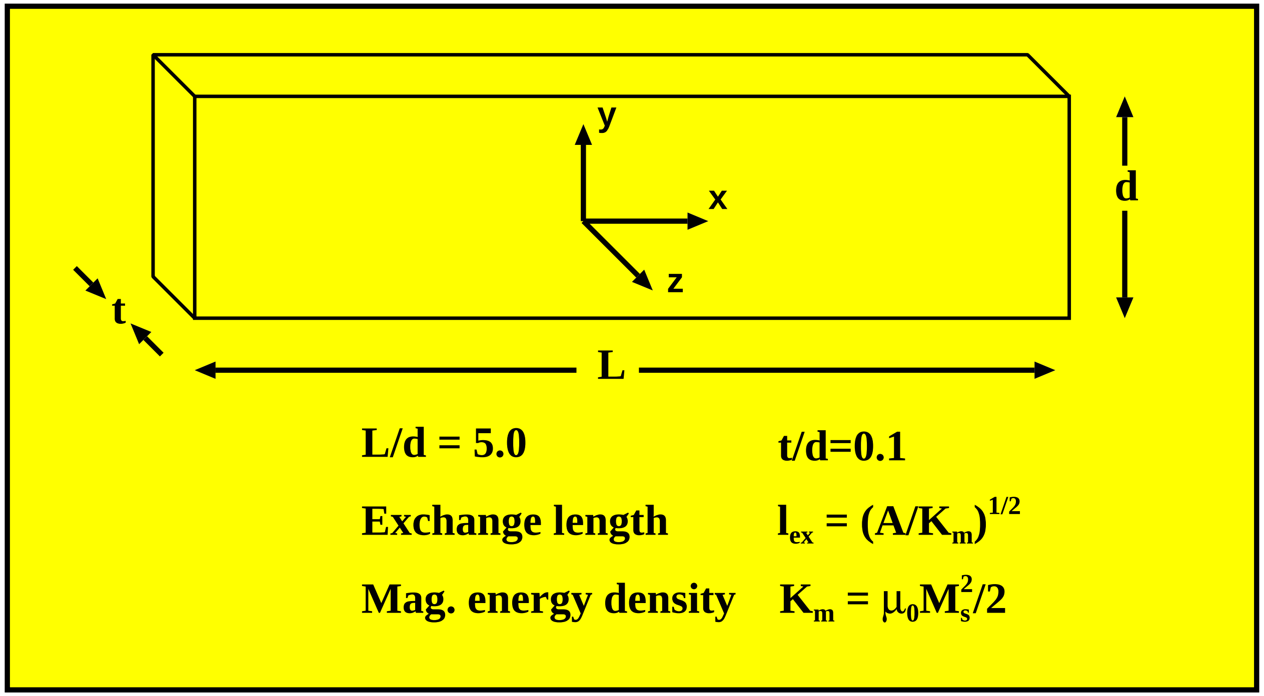
μMAG Standard Problem #1



μMAG Standard Problem #1

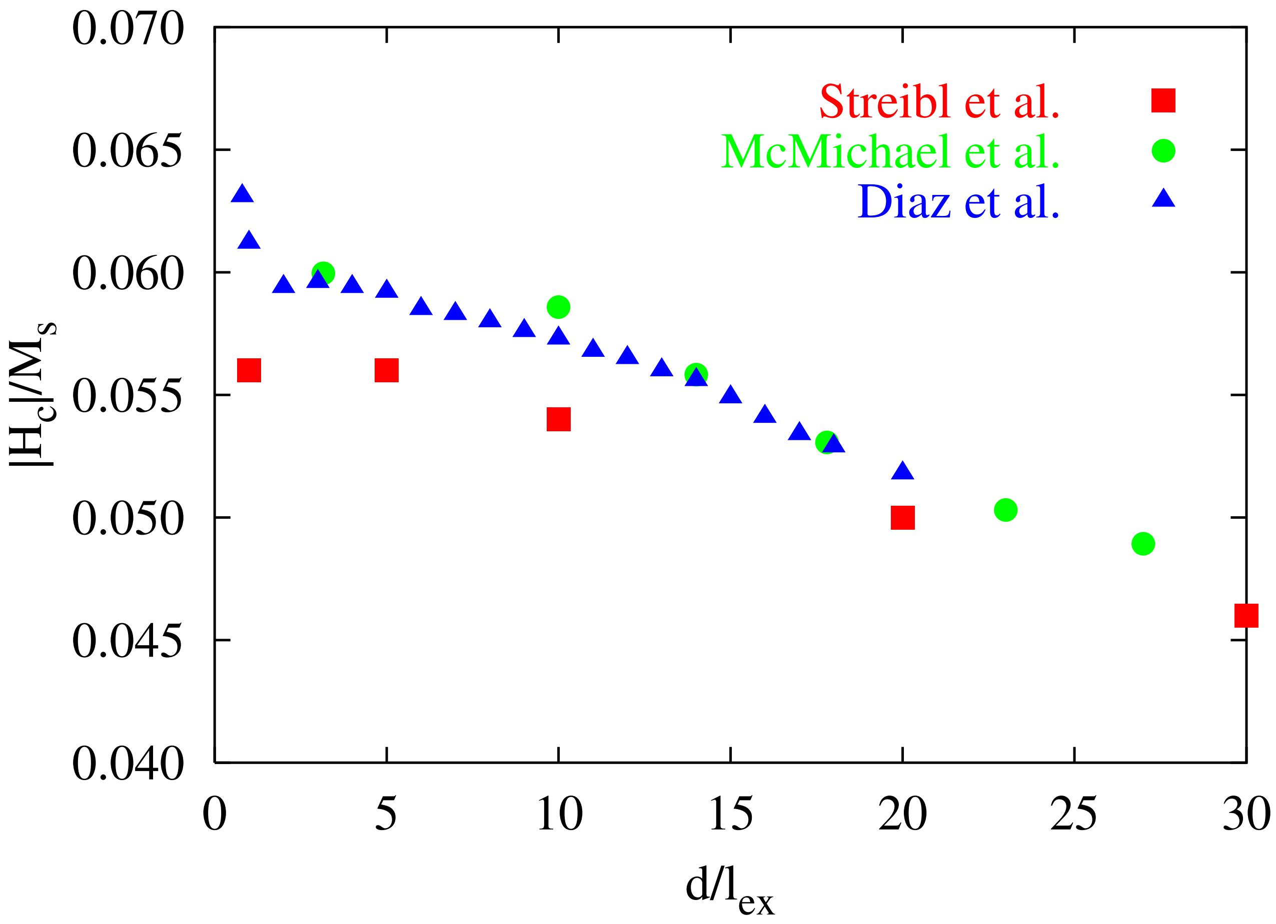


μMAG Problem #2

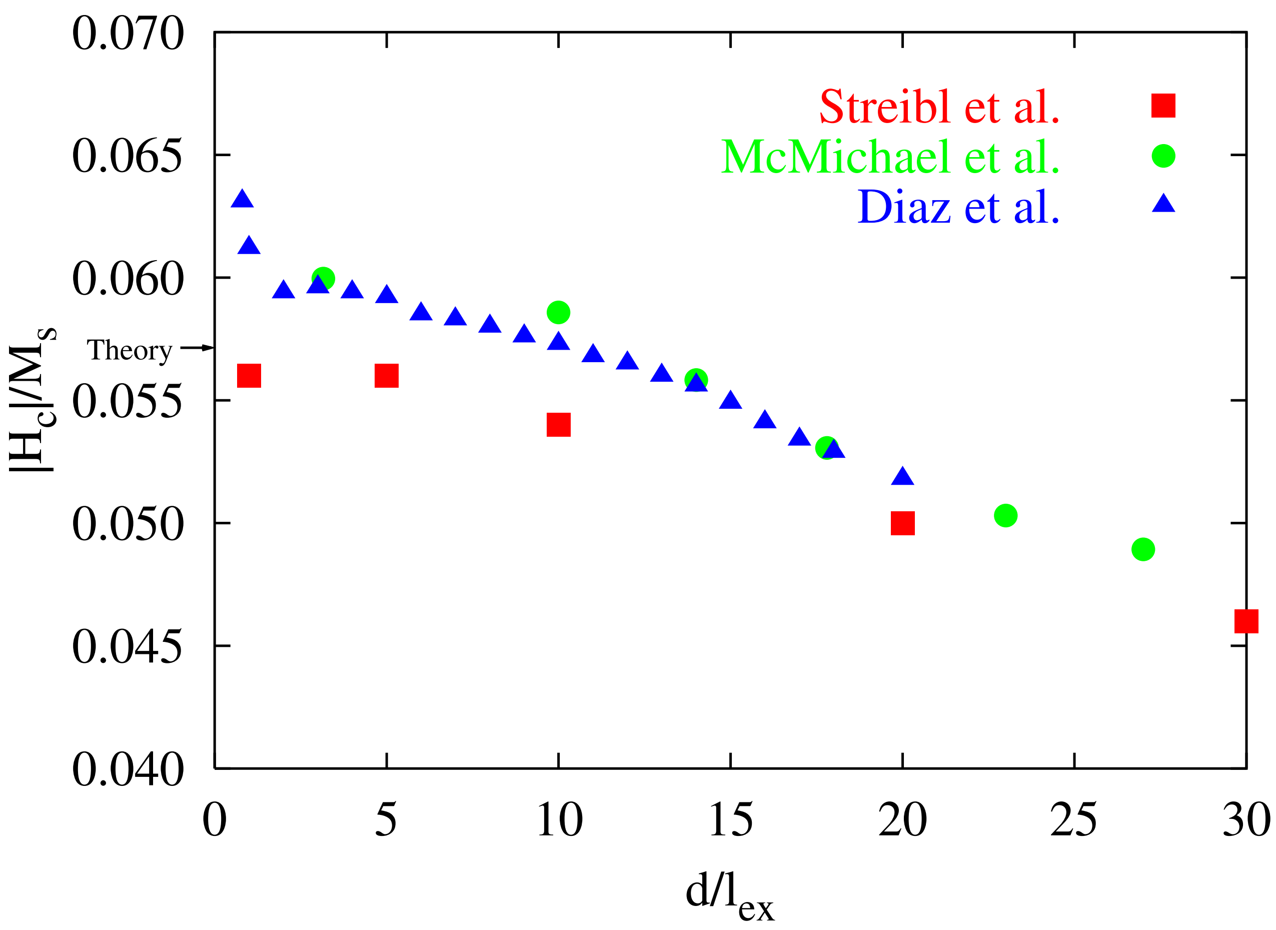


Find coercivity and remanence
as function of $d/l_{\rm ex}$.

μMAG Problem #2

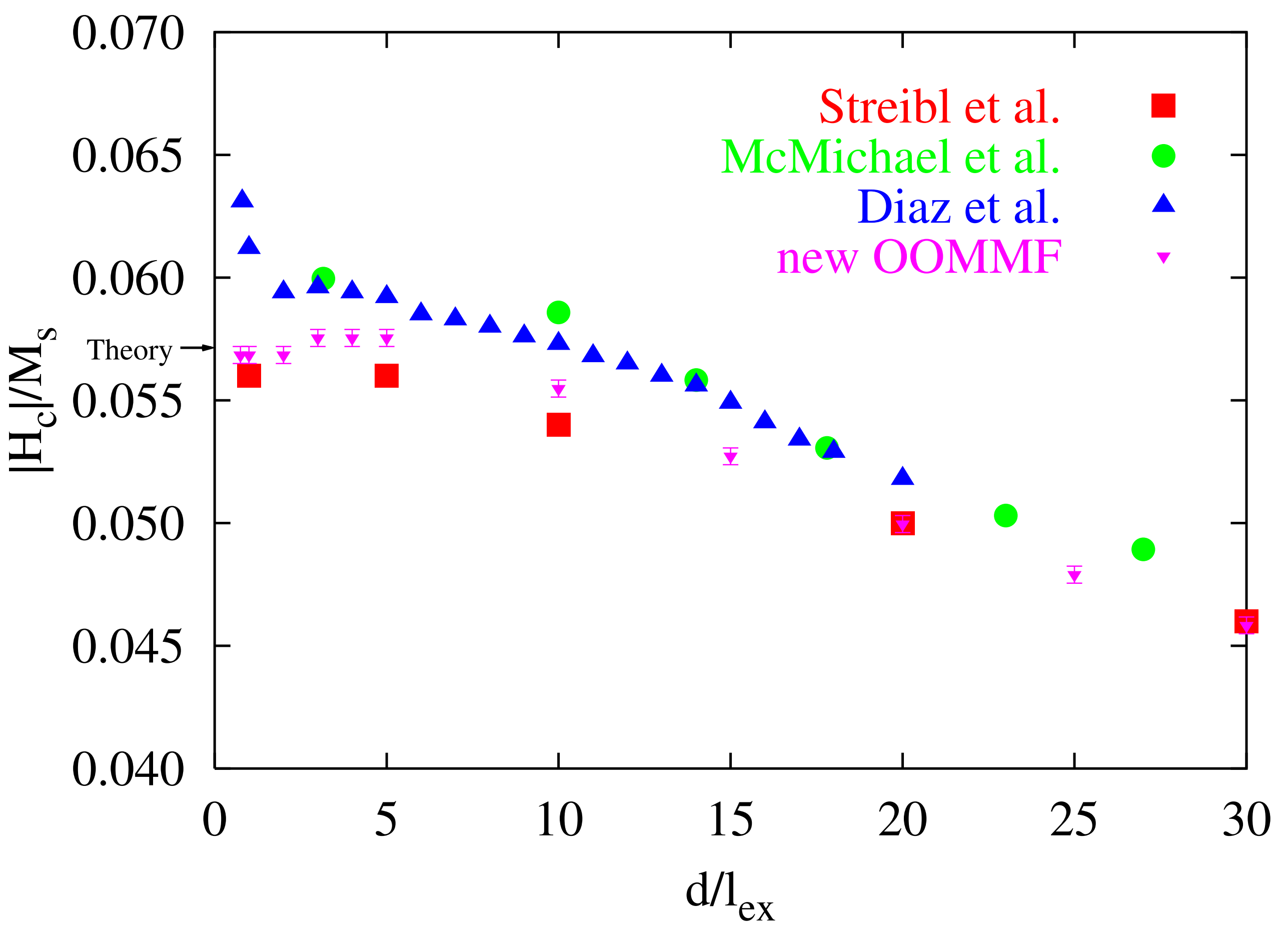


μMAG Problem #2

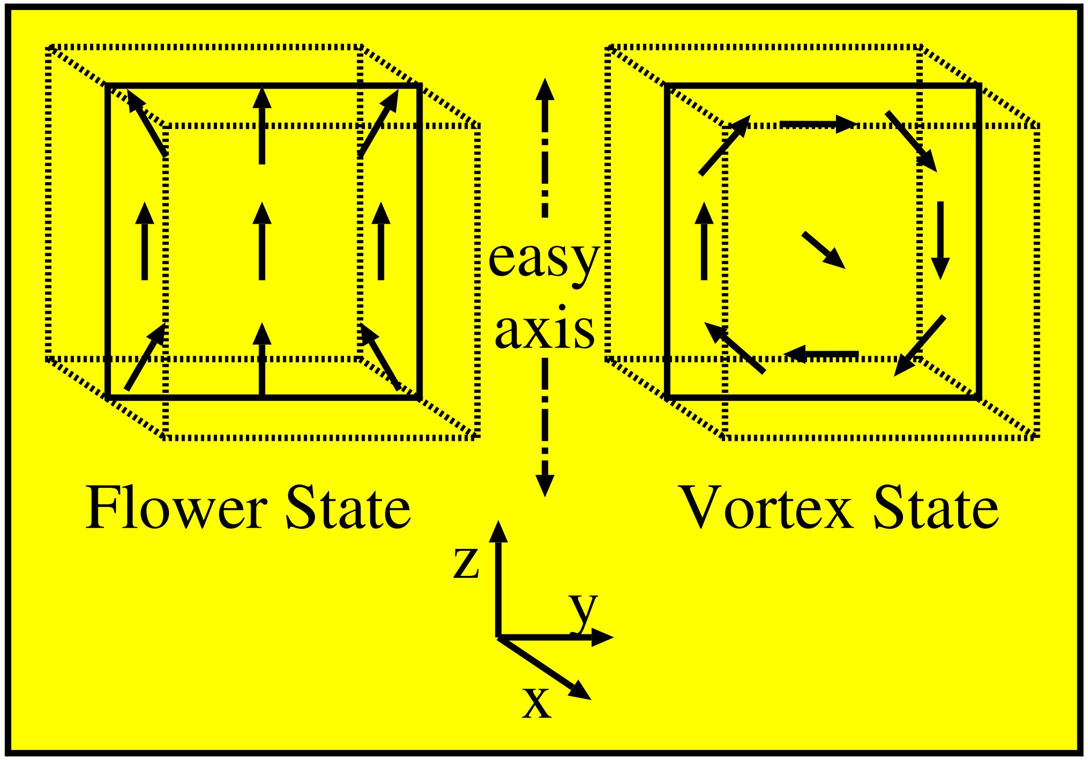


M. J. Donahue and D. G. Porter, *IEEE Trans. Magn.*, **38**, 2468 (2002).

μMAG Problem #2

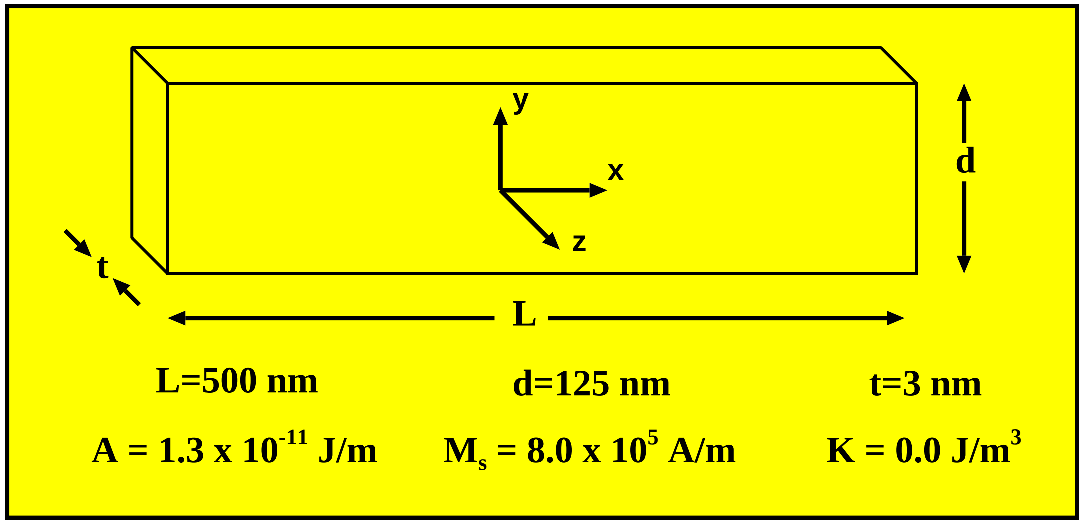


μMAG Problem #3



Compare energies for 2 (3?) states
as function of cube size.

μMAG Problem #4

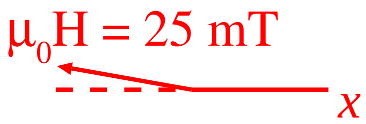


LLG Dynamics: plot M vs. t for two cases

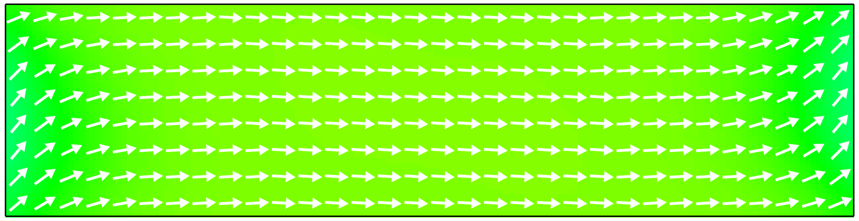
- Field 1: 25 mT, $170°$ from $+x$ axis
- Field 2: 36 mT, $190°$ from $+x$ axis

Problem #4a

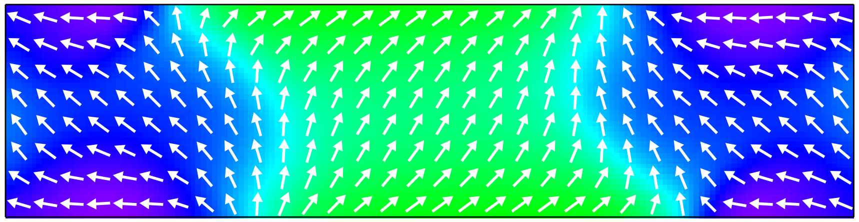
Time



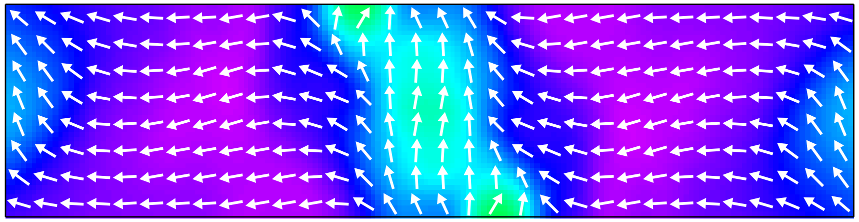
0 ps



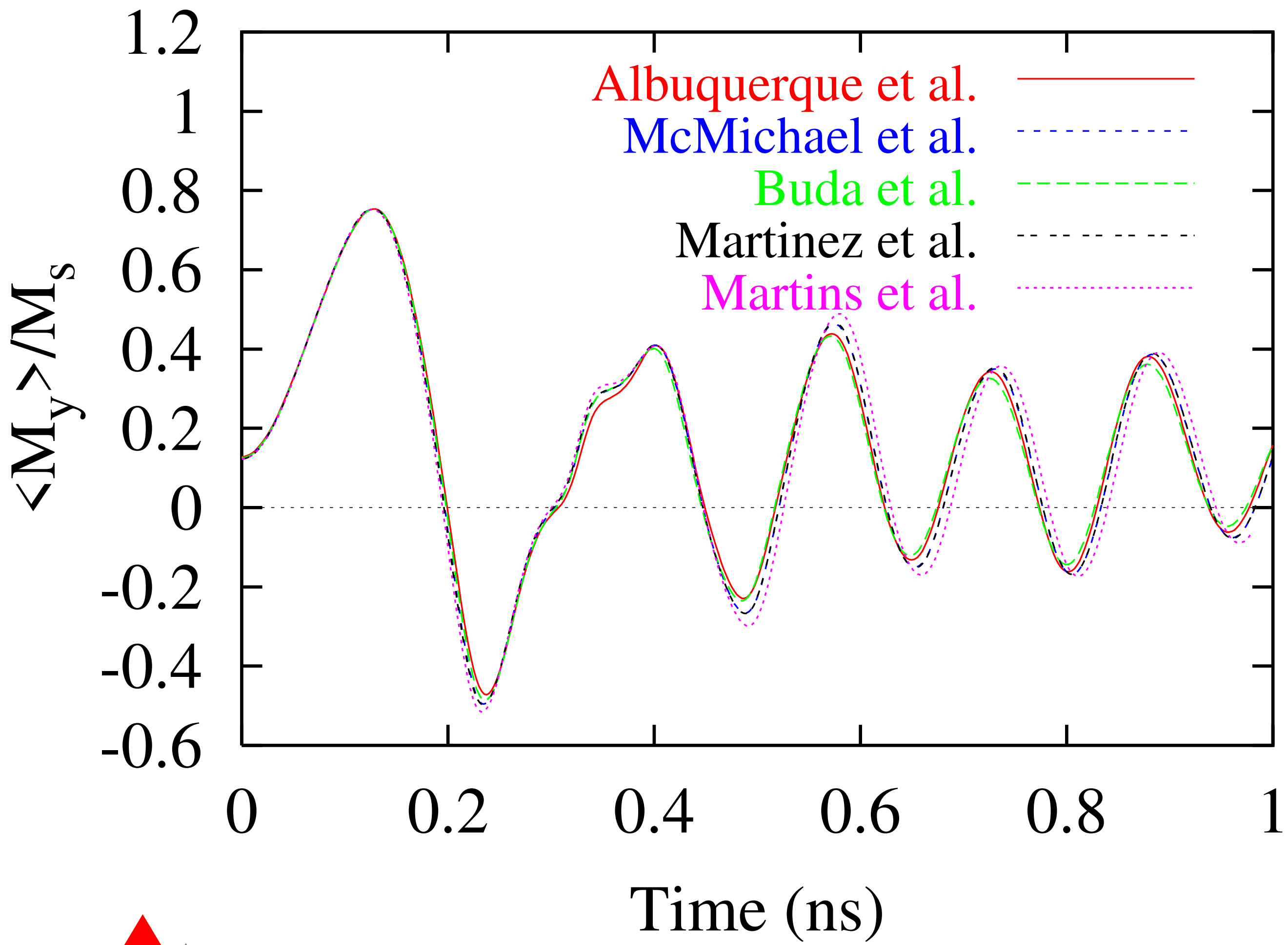
150 ps



350 ps

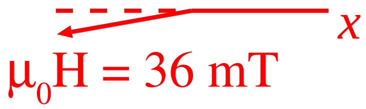


Field 1, 170° *from* $+x$ *axis*

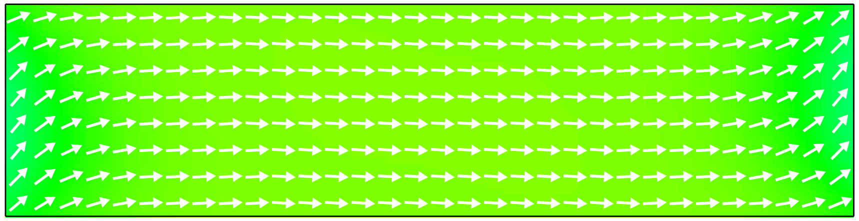


Problem #4b

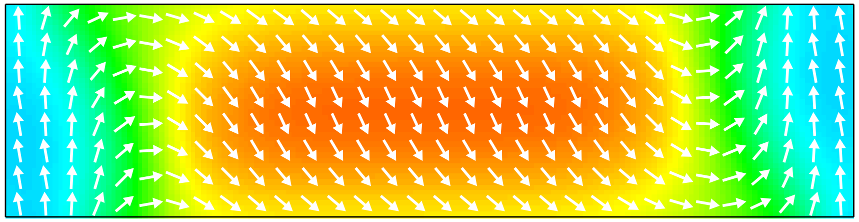
Time



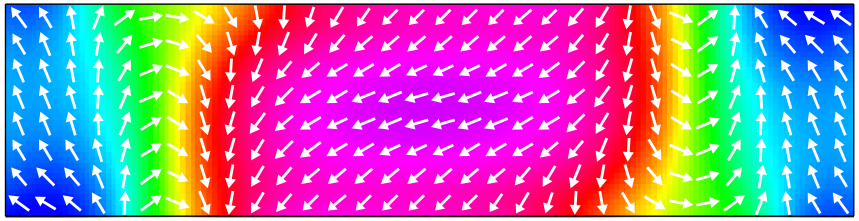
0 ps



100 ps



150 ps

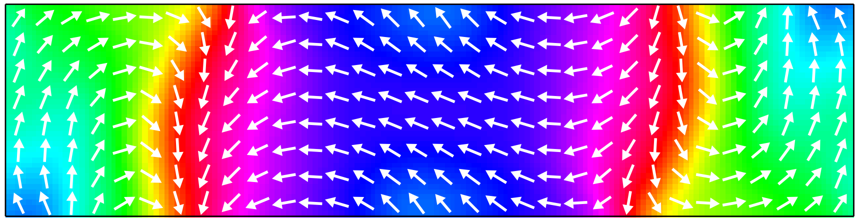


Problem #4b

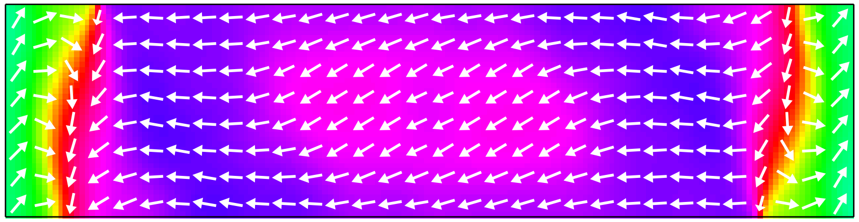
Time

$\mu_0 H = 36$ mT

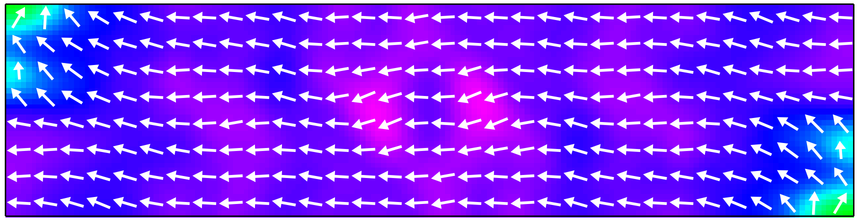
350 ps



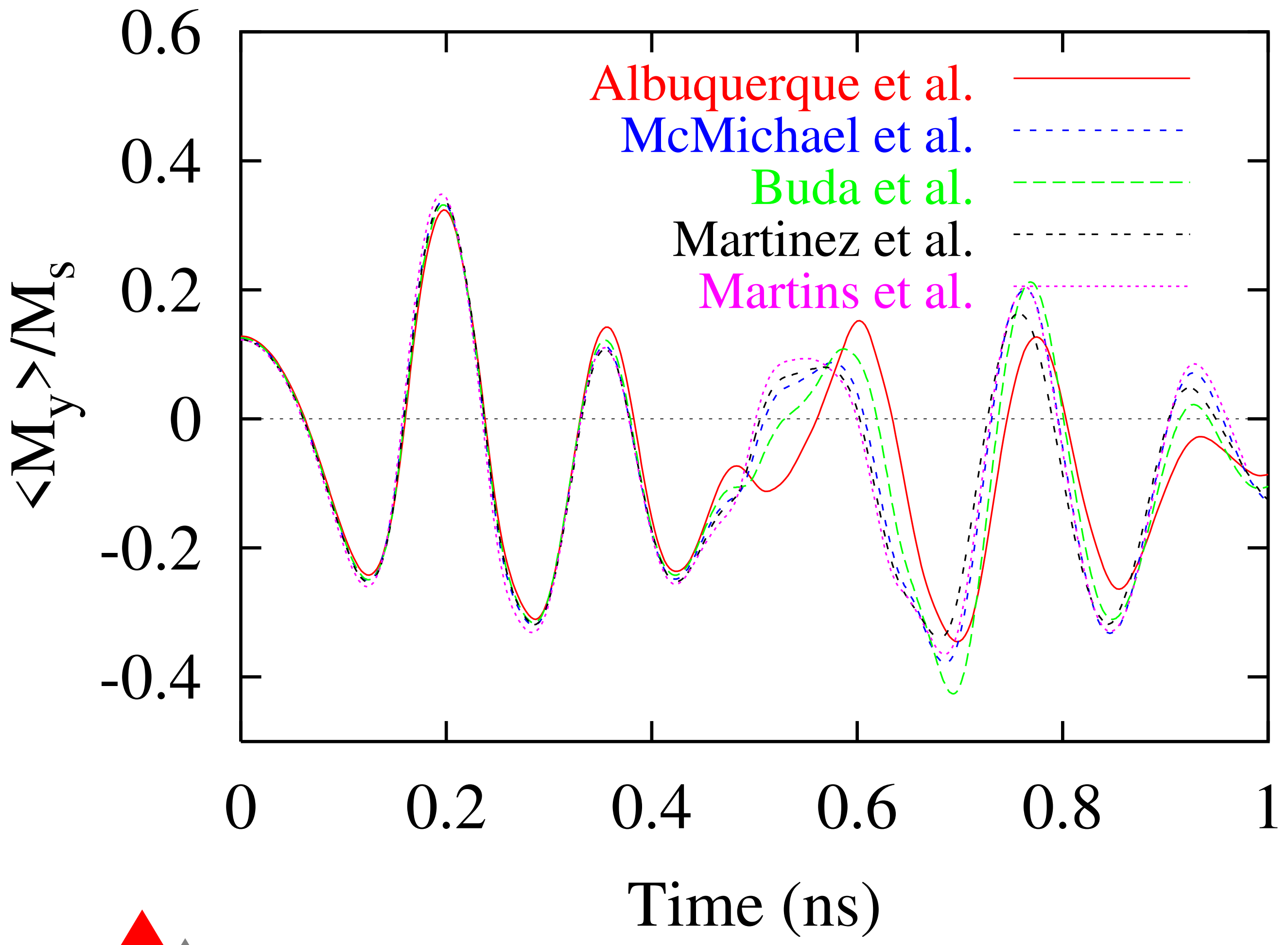
450 ps



750 ps



Field 2, 190° *from* $+x$ *axis*



Standard Problem Ideals

- Small and topical
- Tractable for majority of community
- Test single aspect
- Watch out for symmetry breaking and other instabilities
- One part easy (get agreement)

Future Standard Problems

- Thermal effects
  - LLG + thermal field (ns)
  - Long term stability (seconds – years)
- Boundary effects

Public Code

**Portable, extensible,
public domain
programs & tools
for micromagnetics**

`http://math.nist.gov/oommf`

- Graphical User Interface
- Windows and Unix
- Binaries and source code
- Tcl/Tk and C++ based modular architecture

- 200 page user's manual
- 1000+ downloads in 2001
- 1500+ downloads in 2002
- 1800+ downloads in 2003
- 1200+ downloads through May 2004

Publications using OOMMF in peer-reviewed journals:

Year	Count
$\leq$2000	13
2001	14
2002	38
2003	36
Total	101

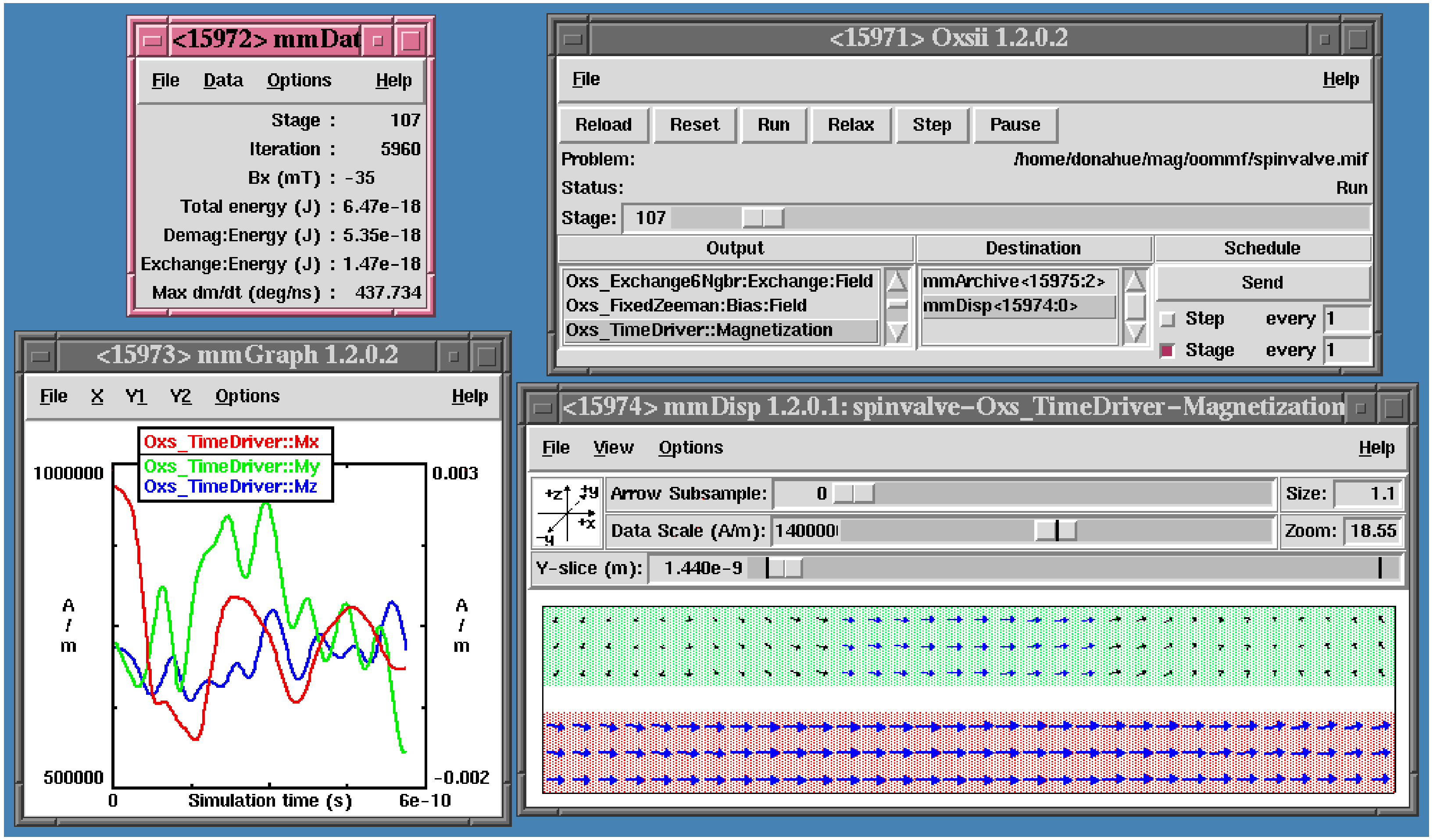
Requirements

- Unix/X, Windows 95+, Mac OS X
- Tcl/Tk 7.6/4.2+ (8.0+ recommended)
  Tcl Developer Xchange: http://www.tcl.tk
- TCP/IP networking on localhost
- RAM: 32 MB (minimum)
- Disk space: 25/80 MB
- Modern C++ compiler to build from source

Testbed Systems

Platform	Compilers
Alpha/Linux	Compaq C++, Gnu gcc
Alpha/Tru64	Compaq C++ (cxx)
HP-UX	aCC
Intel/Linux	Gnu gcc, Intel C++, Portland Group
Intel/Windows	Microsoft Visual C++, Intel C++, Cygwin gcc, Borland C++
MIPS/IRIX 6 (SGI)	MIPSpro C++, Gnu gcc
SPARC/Solaris	Sun Workshop C++, Gnu gcc

Public Code



<15972> mmDat
File Data Options Help
Stage : 107
Iteration : 5960
Bx (mT) : -35
Total energy (J) : 6.47e-18
Demag:Energy (J) : 5.35e-18
Exchange:Energy (J) : 1.47e-18
Max dm/dt (deg/ns) : 437.734
<15971> Oxsii 1.2.0.2
File Help
Reload Reset Run Relax Step Pause
Problem: /home/donahue/mag/oommf/spinvalve.mif
Status: Run
Stage: 107
Output Destination Schedule
Oxs_Exchange6Ngbr:Exchange:Field
Oxs_FixedZeeman:Bias:Field
Oxs_TimeDriver::Magnetization
mmArchive<15975:2>
mmDisp<15974:0>
Send
Step every 1
Stage every 1
<15973> mmGraph 1.2.0.2
File X Y1 Y2 Options Help
Oxs_TimeDriver::Mx
Oxs_TimeDriver::My
Oxs_TimeDriver::Mz
1000000
500000
0.003
-0.002
A/m
0
Simulation time (s)
6e-10
<15974> mmDisp 1.2.0.1: spinvalve-Oxs_TimeDriver-Magnetization
File View Options Help
Arrow Subsample: 0
Size: 1.1
Data Scale (A/m): 140000
Zoom: 18.55
Y-slice (m): 1.440e-9

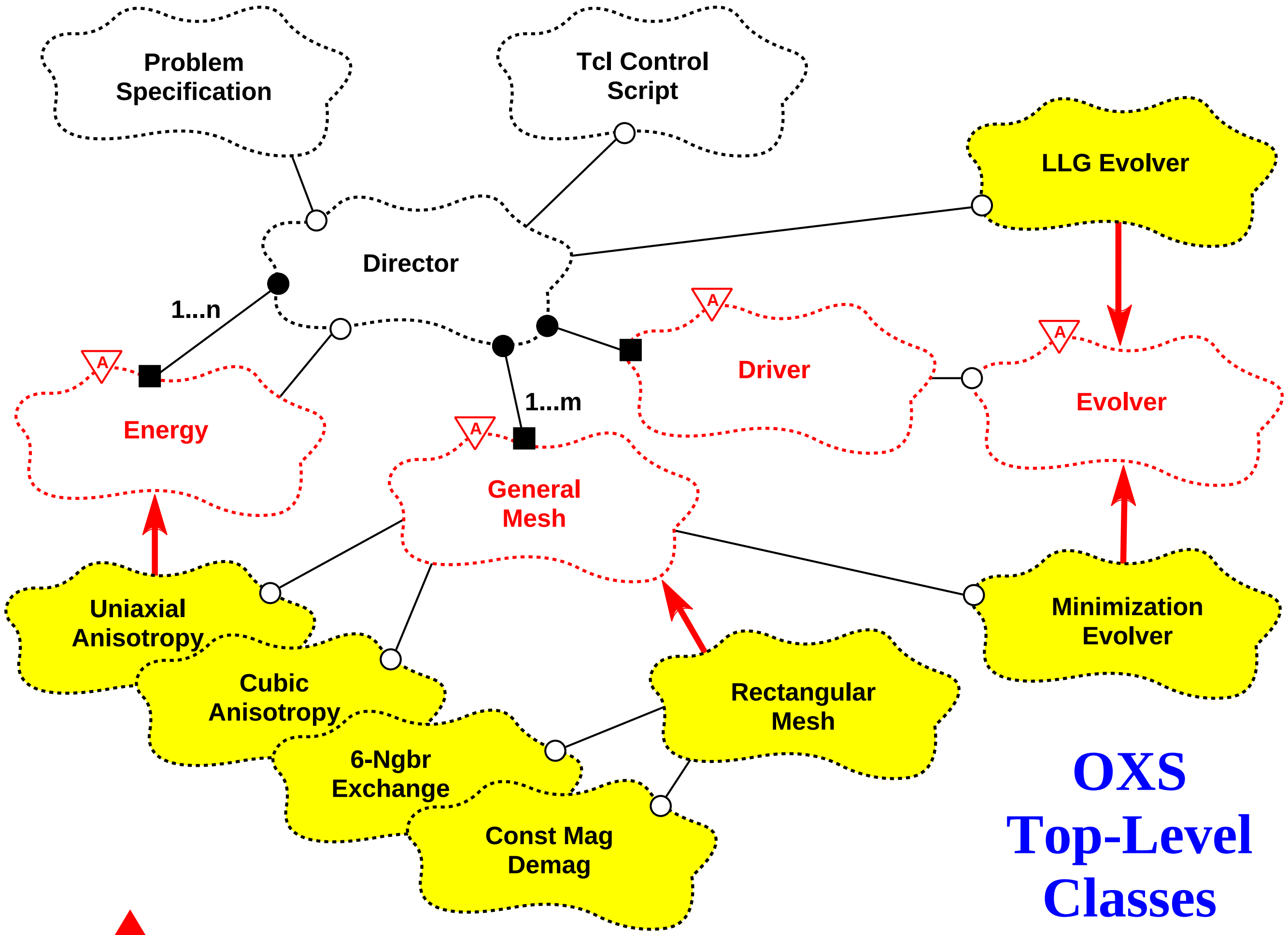
Extending OXS: Input scripts

- Input problem specification (MIF) is Tcl code
- Adjust:
  - material parameters
  - part geometry
  - initial magnetization
  - applied fields (time & space)

Sample MIF file

```
# MIF 2.1
Specify Oxs_BoxAtlas:atlas {
  xrange {0 300e-9}  yrange {0 200e-9}  zrange {0  16e-9}
}
Specify Oxs_RectangularMesh:mesh {
  cellsize {5e-9 5e-9 4e-9}
  atlas :atlas
}
Specify Oxs_UniformExchange { A  8E-12 }
proc Vortex { x y z } {
    set xrad [expr $x-0.5]
    set yrad [expr $y-0.5]
    set normsq [expr $xrad*$xrad+$yrad*$yrad]
    if {$normsq <= 0.0125} {return "0 0 1"}
    return [list [expr -1*$yrad] $xrad 0]
}
...
```

OOMMF eXtensible Solver



OXS Top-Level Classes

OXS Extension Modules

MESHES

rectangularmesh

ENERGY OBJECTS

uniaxialanisotropy
cubicanisotropy
demag
uzeeman
stagezeeman
transformzeeman
scriptuzeeman

uniformexchange
exchange6ngbr
exchangeptwise
randomsiteexchange
twosurfaceexchange
fixedzeeman

OXS Extension Modules

DRIVERS

timedriver   mindriver

EVOLVERS

Time evolvers

euler   rungekutta

Minimization evolvers

conjugategradient

OXS Extension Modules

INITIALIZERS

Scalar fields

atlas
random
uniform
scriptorient
affinetransform
linear
script
vecmag
affineorient

Vector fields

atlas
planerandom
script
scriptorient
affinetransform
file
random
uniform
affineorient

External package support

- CVODE (evolver)
- FFTW (demag)
- thetaevolve (thermal module)

Extending OXS: New Ext classes

1. Create Oxs_Ext class file
    - Initializer
    - Energy & field routine
2. Run pimake
3. Create input file

Oxs_Ext initializer

```
// Constructor
Oxs_HexAnisotropy::Oxs_HexAnisotropy(
  const char* name,     // Child instance id
  Oxs_Director* newdtr, // App director
  const char* argstr)   // MIF input block parameters
  : Oxs_Energy(name,newdtr,argstr), mesh_id(0)
{
  // Process arguments
  OXS_GET_INIT_EXT_OBJECT("K1",Oxs_ScalarField,K1_init);
  OXS_GET_INIT_EXT_OBJECT("K2",Oxs_ScalarField,K2_init);
  OXS_GET_INIT_EXT_OBJECT("K3",Oxs_ScalarField,K3_init);
  OXS_GET_INIT_EXT_OBJECT("K4",Oxs_ScalarField,K4_init);
  OXS_GET_INIT_EXT_OBJECT("axis1",Oxs_VectorField,axis1_init);
  OXS_GET_INIT_EXT_OBJECT("axis2",Oxs_VectorField,axis2_init);
  VerifyAllInitArgsUsed();
}
```

Oxs_Ext energy

```
void Oxs_HexAnisotropy::GetEnergy
(const Oxs_SimState& state,Oxs_EnergyData& oed) const {
<SNIP: Boilerplate initialization>
  for(UINT4m i=0;i<size;++i) {
    const ThreeVector&  m = spin[i];
    const REAL8m k1 = K1[i];    const REAL8m k2 = K2[i];
    const REAL8m k3 = K3[i];    const REAL8m k4 = K4[i];
    const ThreeVector& u1 = axis1[i];
    const ThreeVector& u2 = axis2[i];
    ThreeVector mxu1 = m;    mxu1 ^= u1;
    REAL8m sintheta2 = mxu1.MagSq();
    REAL8m sintheta6 = sintheta2*sintheta2*sintheta2;
    REAL8m costheta = m*u1;
    REAL8m cosphi = m*u2;
    REAL8m cosphi2 = cosphi*cosphi;
    REAL8m cos6phi = ((32*cosphi2-48)*cosphi2+18)*cosphi2-1;
```

Oxs_Ext (cont.)

```
      // Compute energy at cell i
      energy[i] = (((k4*cos6phi+k3)*sintheta2+k2)*sintheta2+k1)
        *sintheta2;

      // Compute field at cell i
      REAL8m Hcoef1 = (((k4*cos6phi+k3)*3*sintheta2+2*k2)*sintheta2+k1)
        *2*costheta;
      REAL8m Hcoef2 = -12*k4*cosphi*sintheta6
        *((cosphi2-1)*16*cosphi2+3);
      field[i] = Hcoef1*u1;
      field[i] += Hcoef2*u2;
      field[i] *= field_mult;
    }
  }
```

Sample input file

```
...
Specify Oxs_HexAnisotropy {
  K1  -4.5e3
  K2   100
  K3   100
  K4   1e4
  axis1 {0 0 1}
  axis2 {1 0 0}
}
...
```

Future work

- Thermal effects
- Multiple grids
- Singular points

Summary

- Standard problems:
  - Useful mainly to code developers
  - Encourage dialog
  - Identify problems
- Public code:
  - Reference implementation for developers
  - Starting point for new developers
  - Used by broad community for published research (transparency)

References

- μMAG:
  http://www.ctcms.nist.gov/~rdm/mumag.org.html
- OOMMF:
  http://math.nist.gov/oommf/
- OOMMF User's Guide, Version 1.0
  M. J. Donahue and D. G. Porter, **NISTIR 6376**, NIST, Gaithersburg, MD (Sept 1999).