Spinsim Simulator Documentation

Written 2010-02-04 by S. Moyerman, I. Yulaev
University of California, San Diego
ECE Dept.

Table of Contents

Overview

Spinsim simulates the magnetization of a free-pinned-layer structure along a variety of externally applied H-field and current density conditions. It provides a graphical user interface and allows plotting of results. Additionally, many of the parameters used in the calculation may be specified.

Licensing Terms

This program is distributed under some sort of license. It is not public domain.

Terminology

Some terms in this document are used interchangably, and thus require disambiguation.

The terms “Spinsim” and “Spinsim GSL” both reference the same project, namely, this project.

“Iteration” is used to refer to a time-independent current and applied field value pair, with which the magnetization of the structure is calculated. Typically, the spinsim simulator output will consist of the magnetization for a grid of current/applied field pairs. Each point on the grid is an “iteration”.

Dependencies and Required Package

Some packages are required to compile and run this program. It has been tested under Linux x86-64. All of the packages required may be obtained via Debian's Synaptic Package Manager. The following is a partial list of required packages:

- gsl-bin
- python-qt4
- python-matplotlib
- python-qt4-dev (*)
- libgsl0-dev (*)
- libgsl0ldbl (*)
- pyqt4-dev-tools (*)
- qt4-designer (*)

(*) denotes required for development only

How to Compile

There are two parts to the Spinsim program. The basic engine is a compiled C program that generates the output in numerical format. It may be run from the command line, with arguments specified there. There is also a GUI written in Python/PyQt, which provides a graphical interface for running the program, and via matplotlib, allows plotting of the results.

C Code: Run “gcc -lgsl -lgslcblas -lm -o spinsim spinsim.c” from the command line.

Python: The GUI may be run by typing “python spinsim_gui.py”.

If you have been developing the GUI file qt_ui.ui, it must be re-compiled. Run pyuic4 qt_ui.ui > qt_ui.py. *This is not necessary for most users.*

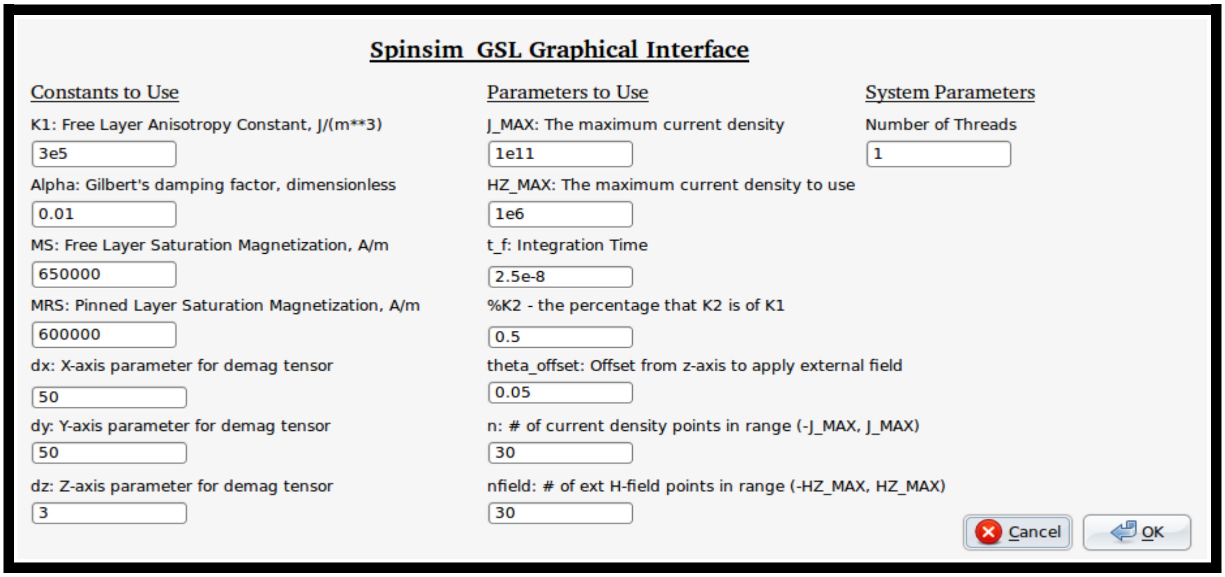
Contact

The principal contacts for support regarding this project are Stephanie Moyerman (smoyerman@ucsd.edu) and Ivan Yulaev (iyulaev@ucsd.edu).

How To Use (GUI)

Note: As of 2010-01-23, the GUI does not support time-dependent current and magnetic field functions. These are only usable through the command line interface.

The simpler way to use this program is via the GUI. In the spinsim directory, type “python spinsim_gui.py”. You should see the following dialog:



Spinsim GSL Graphical Interface

Constants to Use

K1: Free Layer Anisotropy Constant, J/(m**3)
3e5

Alpha: Gilbert's damping factor, dimensionless
0.01

MS: Free Layer Saturation Magnetization, A/m
650000

MRS: Pinned Layer Saturation Magnetization, A/m
600000

dx: X-axis parameter for demag tensor
50

dy: Y-axis parameter for demag tensor
50

dz: Z-axis parameter for demag tensor
3

Parameters to Use

J_MAX: The maximum current density
1e11

HZ_MAX: The maximum current density to use
1e6

t_f: Integration Time
2.5e-8

%K2 - the percentage that K2 is of K1
0.5

theta_offset: Offset from z-axis to apply external field
0.05

n: # of current density points in range (-J_MAX, J_MAX)
30

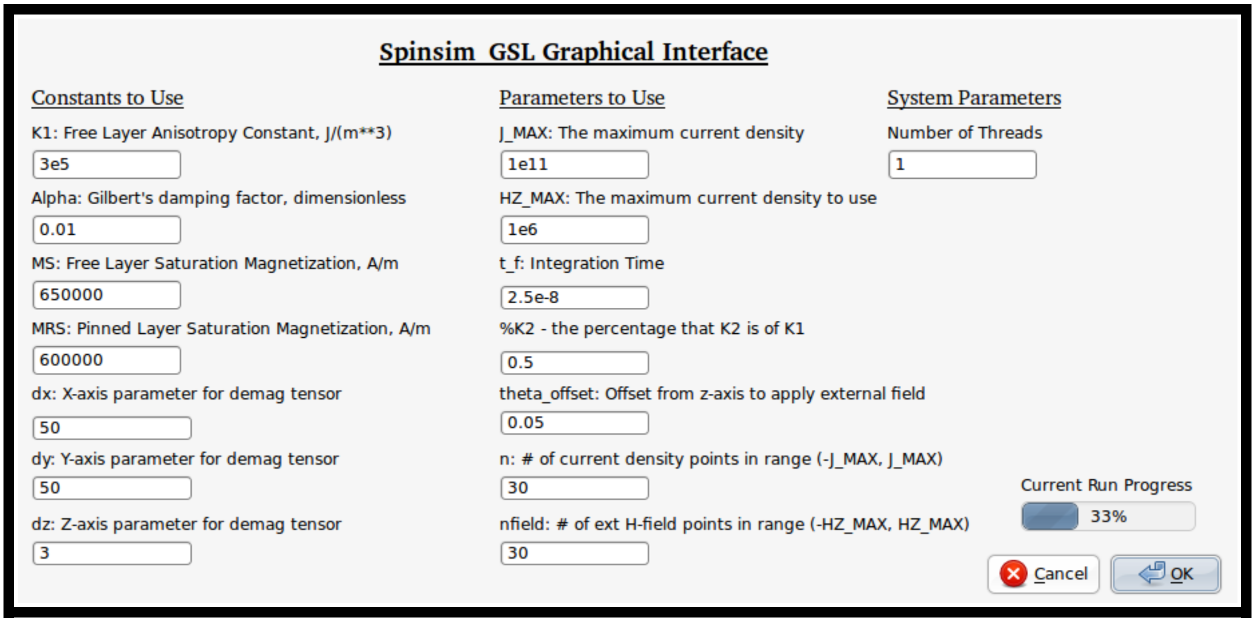
nfield: # of ext H-field points in range (-HZ_MAX, HZ_MAX)
30

System Parameters

Number of Threads
1

The provided parameters are “suggested” parameters for a trial run. You may set the parameters to whatever you want. On a modern machine, the program will take about a tenth of a second for each current density – H-field pair. So, a 30x30 run will take about two minutes. Make sure to set n and nfield to something reasonable.

Then, hit OK. A progress bar will appear, as below



Spinsim GSL Graphical Interface

Constants to Use

K1: Free Layer Anisotropy Constant, J/(m**3)
3e5

Alpha: Gilbert's damping factor, dimensionless
0.01

MS: Free Layer Saturation Magnetization, A/m
650000

MRS: Pinned Layer Saturation Magnetization, A/m
600000

dx: X-axis parameter for demag tensor
50

dy: Y-axis parameter for demag tensor
50

dz: Z-axis parameter for demag tensor
3

Parameters to Use

J_MAX: The maximum current density
1e11

HZ_MAX: The maximum current density to use
1e6

t_f: Integration Time
2.5e-8

%K2 - the percentage that K2 is of K1
0.5

theta_offset: Offset from z-axis to apply external field
0.05

n: # of current density points in range (-J_MAX, J_MAX)
30

nfield: # of ext H-field points in range (-HZ_MAX, HZ_MAX)
30

System Parameters

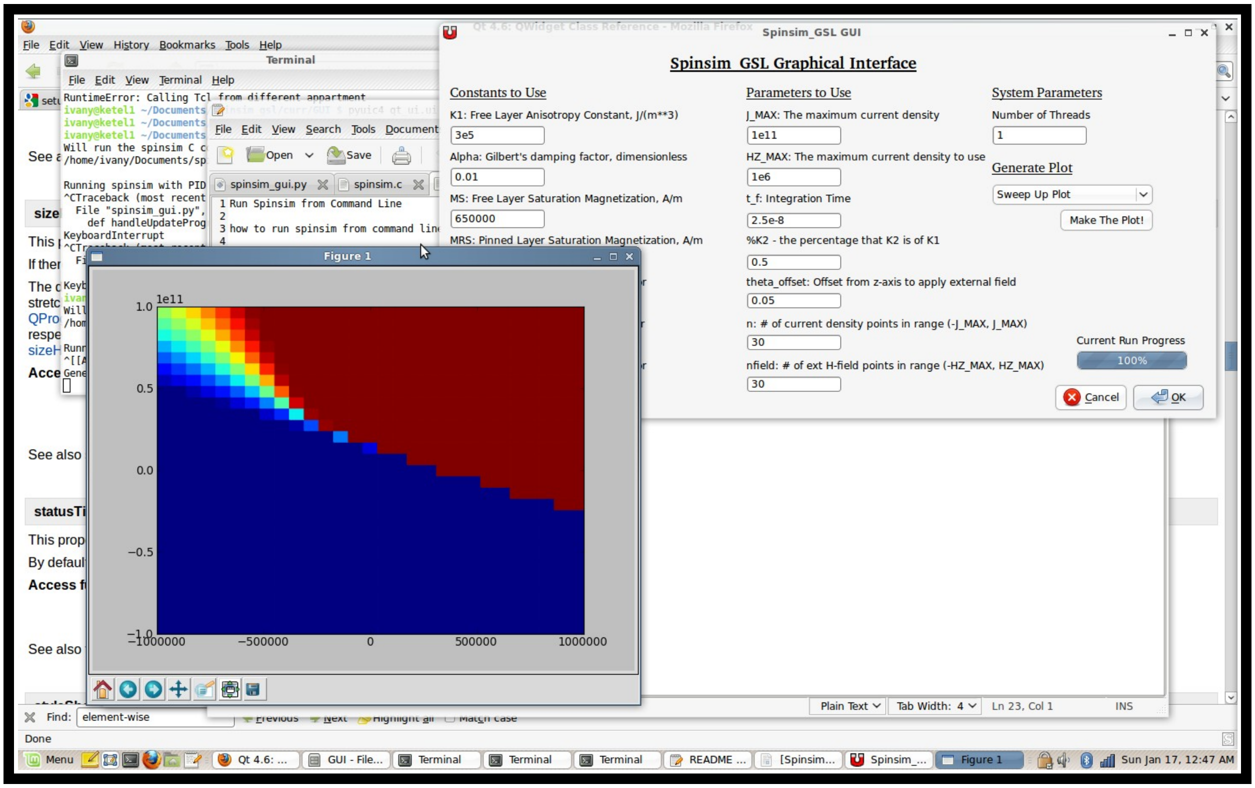
Number of Threads
1

Current Run Progress
33%

Allow it to finish running. Then, a selection list will appear along with a button, allowing you to plot the data. The data will also be put into four files in the current directory:

- magsweepup_output: output for the up-sweeping magnetization
- magsweepdown_output: output for the down-sweeping magnetization
- magsweep_output_j: The list of current density conditions used
- magsweep_output_hz: The list of external field conditions used

Also, you may plot the data by pressing the “Make the Plot” button. The output should appear as below. Note that every time you hit “Make the Plot” it will also output a high-res PNG of the plot you just made to the same folder as the spinsim_gui.py file.



```
RuntimeError: Calling Tcl from different appartment
ivany@ketel1 ~/Documents
ivany@ketel1 ~/Documents
ivany@ketel1 ~/Documents
Will run the spinsim C c
/home/ivany/Documents/sp

Running spinsim with PID
^CTraceback (most recent
  File "spinsim_gui.py",
    def handleUpdateProg
KeyboardInterrupt
```

```
1 Run Spinsim from Command Line
2
3 how to run spinsim from command line
4
```

Spinsim_GSL Graphical Interface

Constants to Use

K1: Free Layer Anisotropy Constant, J/(m**3): 3e5

Alpha: Gilbert's damping factor, dimensionless: 0.01

MS: Free Layer Saturation Magnetization, A/m: 650000

MRS: Pinned Layer Saturation Magnetization, A/m

Parameters to Use

J_MAX: The maximum current density: 1e11

HZ_MAX: The maximum current density to use: 1e6

t_f: Integration Time: 2.5e-8

%K2 - the percentage that K2 is of K1: 0.5

theta_offset: Offset from z-axis to apply external field: 0.05

n: # of current density points in range (-J_MAX, J_MAX): 30

nfield: # of ext H-field points in range (-HZ_MAX, HZ_MAX): 30

System Parameters

Number of Threads: 1

Generate Plot: Sweep Up Plot

Current Run Progress: 100%

In the plot window, you can use the controls (lower left) to move the plot and zoom in on certain areas. After zooming in on a certain area you can click “Limits from Plot” button *(not shown in above screenshot)*, this will extract the limits from the LAST plot that was resized, IF it was resized from the default values. The purpose of this is to allow re-plotting of a region of interest in a plot, for finer resolution when observing some part of the larger plot.

How to Use (Command Line)

The command line interface allows the user to call the spinsim engine directly from the command line, and pass the parameters in via command line arguments rather than using a GUI. There is not much of a speed advantage (the GUI adds very little overhead) but this interface may be useful for scripting.

The spinsim executable expects parameters to be passed in via command line arguments. The order of the arguments is

(1) K1 - Free layer anisotropy constant, J/m^3
(2) Alpha - Gilbert's damping factor, dimensionless
(3) Ms - Free layer saturation magnetization, A/m
(4) Mrs - Pinned layer saturation magnetization, A/m
(5) dx - x-axis parameter for demagnetization tensor
(6) dy - y-axis parameter for demagnetization tensor
(7) dz - z-axis parameter for demagnetization tensor
(8) Min j – The minimum value to use for current density
(9) Max j - the maximum DC value to use for current density, A/m^2
(10) Min Hz - the minimum value to use for externally-applied field
(11) Max Hz - the maximum value to use for externally-applied field
(12) Integration time - the integration time to run until (in seconds)
(13) %K2 - the percentage that the 2nd order damping factor is of the first-order
(14) thetaoff - the offset (in degrees) of the applied magnetic field vs the z-axis of the structure
(15) n - the number of applied current values to use, spaced evenly between (-Max j) and Max j
(16) nfield - the number of external field values to use, spaced evenly between -Max Hz and Max Hz
(17) Number of threads - the number of computation threads to use. Recommended to set this to the number of processing threads your computer can run concurrently.
(18) Bit-wise OR of time-dependent current density functions to use - see section “how to enter and use time-dependent functions”
(19) Bit-wise OR of time-dependent external field functions to use - see section “how to enter and use time-dependent functions”

An example of this syntax, which uses 2 threads and only time-independent applied current and external field, is

```
./spinsim 3e5 0.01 650000 600000 50 50 3 -1e11 1e11 -1e6 1e6 2.5e-8 0.5 0.05 30 30 2 1 1
```

In this syntax we use 3e5 as the free layer anisotropy constant, 650000 as free layer saturation magnetization, 600000 as pinned layer saturation magnetization, (50,50,3) for (dx, dy, dz), -1e11 for minimum current density, 1e11 for maximum current density, -1e6 for minimum applied field, 1e6 for maximum applied field, 2.5e-8 seconds as the integration time, 0.5 as the percentage that K2 is of K1, 0.05 as the offset in degrees of the applied magnetic field vs the z axis, calculating over 30 external field and 30 current density values, using 2 processing threads, and only using time-independent values for field and current.

Manual Entry of Time-Dependent Field and Current Parameters

In this section, we will describe how to enter time dependent field parameters. In the default case, the spinsim engine will use fixed applied field and applied current conditions for calculating magnetization for a given iteration. It is possible, however, to make applied field and applied current time-dependent so that, for example, it is possible to simulate the effect of passing an AC current through the structure.

There are two time-dependent waveform types defined for current and field – sine wave and user-defined points. These are both specified in external files and loaded by the spinsim program at runtime. Parameters are specified in a file named “fun_params”, located in the same directory as the spinsim executable.

Sine Wave Specification

Sine waves can be of two types, DC dependent or independent. In the dependent case, the amplitude of the sine wave is given as a scalar multiple of the time-independent waveform being used. In the independent case, the absolute amplitude is given. Both cases also require a frequency to be entered (in Hz) as well as a phase offset (in radians). Example syntax in the fun_params file is

```
j_sin_dep      0.2     1000000000  1.571
```

This line defines a dependent sine wave for current density. Its amplitude will be 20% of the magnitude of the current density used for the given iteration, its frequency will be 1GHz, and it's phase offset will be PI/2.

```
j_sin_indep    1e5     1e9     0.785
```

This line defined an independent sine wave for current density. Its amplitude 10,000 A/m**2, its frequency will be 1GHz, and it's phase offset will be PI/4.

Similar syntax may be used for defining dependent and independent magnetic field density. The keyword at the beginning of the line should be “hz_sin_dep” and “hz_sin_indep” respectively.

User Waveform Specification

User waveforms may be specified using (time, value) pairs, to create arbitrary waveforms for a simulation. The values may be absolute current or field values, or may be scaling values for the current iteration's time-independent external field and current density values.

In the fun_params file, the user waveforms are specified using syntax like the following

```
hz_user_dep   timedep_user_funs/hz_dep_fun_file
```

In this case, the user waveform for the external magnetic field will be loaded from the file timedep_user_funs/hz_dep_fun_file. The file is expected to consist of tab delimited time-value pairs, such as the following:

```
1e-9    0.1
10e-9   0.2
15e-9   0.3
```

Note that in the dependent case, the value is assumed to be a scaling factor for the time-independent current/field value. Other keywords for the fun_params file are hz_user_indep, j_user_dep, j_user_indep. These may be used to specify an independent field waveform, a dependent current waveform, and an independent current waveform, respectively.

Calling Spinsim from the Command Line with Time-Dependent Parameters

The last two command-line argument for the spinsim executable describe what time-dependent functions will be used. The values are treated as bitwise-OR'd numbers, so that any combination of time-independent and dependent waveforms may be used for both current and field. The valid bits for these arguments are

1 – superimpose time-independent current/field value

2 – superimpose sine function that depends on time-independent current/field value

4 – superimpose sine function that doesn't depend on time-independent current/field value

8 – superimpose user function that depends on time-independent current/field value

16 – superimpose user function that doesn't depend on time-independent current/field value

These numbers may be added arbitrarily, to produce any combination of the above. For example, if one wished to use the time-independent current, a sine function that depends on the time-independent current, and a user function that is independent of the time-independent current, one would use as the command line argument

$16 + 2 + 1 = 19$

So, the last command line argument for calling the spinsim function would be 19, as in the following syntax.

```
./spinsim 3e5 0.01 650000 600000 50 50 3 1e11 1e6 2.5e-8 0.5 0.05 30 30 2 1 19
```

Similarly, if one wished to use the same current setup as above but additionally, for applied field, use an absolutely-specified user function and the time-independent field value, one would use

$8 + 1 = 9$

as the second-to-last command line argument for spinsim executable. The syntax would be

```
./spinsim 3e5 0.01 650000 600000 50 50 3 1e11 1e6 2.5e-8 0.5 0.05 30 30 2 9 19
```

Examples

For additional examples of specifying and using time-dependent current and field functions, please see the README file, the fun_params file included with the project, and the files in timedep_user_funs/.

Description of Code in the Project

A explicit description of the code is outside the scope of this document. However, the diagram “spinsim_project_diagram.pdf” describes the interaction of the various modules in this program. For more detailed description of the various functions and modules in this project, it is recommended the developer look at the comments at the declaration of each function.

Diagram Errata: *Spinsim_make_plots() function in spinsim_gui.py is now in a separate file, and is called by spinsim_gui.py via a shell command. Functionality is otherwise the same.*