Monte Carlo Dislocation Simulation Code

07/26/2012

1 Building the code

Requirements:

1. C++ compiler supporting the recent C++11 standard
2. CMake makefile generator tool (www.cmake.org)
3. Boost library (www.boost.org)
4. Optional: ParaView software to visualize the dislocation configurations or create a movie

On the LC machines, make these things available by running

```
use gcc-4.6.1
use cmake
use boost-1.49.0
use paraview-opengl
```

Create an empty directory to build the executable in. From within that build dirctory, run

```
CC=gcc
CXX=g++
cmake <path-to-source>
```

where `<path-to-source>` is the path to the directory "`src`" of the simulation package. After CMake has generated the Makefile, run

```
make -j4
```

to compile the program, which is called `DislocationKMC`.

2 Program usage

Running `DislocationKMC` without any parameters will print the list of available program options and simulation parameters. To run a simulation, the program is invoked as follows:

```
./DislocationKMC --config <parameter_file>
```

where `<parameter_file>` points to a text file that contains all simulation parameters. This file consists of lines of the form `<key>=<value>`. The parameters are documented in the following section.

3 Simulation parameters

Output parameters:

`verbosity` Verbosity level (0-3). Controls the amount of log output generated by the program, ranging from 0 (no output) to 3 (super detailed).

`print_interval` Controls how often the dislocation position is output. This is the number of simulation steps to perform before the dislocation position is computed and output again.

`output_file` The output filename for dislocation configurations. If, for instance, the specified output filename is `disloc.vtk`, then the program will generate a sequence of files `disloc.0.vtk`, `disloc.100.vtk`, `disloc.200.vtk`... depending on the selected output interval.

`output_interval` Controls how often the dislocation configuration is written to a VTK output file.

General parameters:

`line_length` Length of screw dislocation line in units of $[b]$. This must be a positive integer specifying the number of Burgers vectors in the periodic z direction.

`random_seed` Positive integer used to initialize the random number generator.

`simulation_steps` Number of Monte Carlo steps to perform. This controls the length of the simulation run.

Simulation parameters:

`stress_xx`, `stress_yy`, `stress_zz`, `stress_xy`, `stress_xz`, `stress_yz` Components of applied stress tensor [Pa]. Note that only `stress_xz` and `stress_yz` are actually used in the simulation.

`temperature` The simulation temperature [K].

Materials parameters:

`lattice_param` BCC Bravais lattice parameter [Å].

`shear_modulus` The elastic shear modulus of the material [Pa].

`poisson_ratio` Poisson's ratio.

`peierls_stress` The Peierls stress of the material [Pa]. In the current implementation, this is assumed to be the same for all kink directions, which are all of {110} type.

`core_width` The size of the dislocation core region [Å]. This parameter is used in the non-singular elasticity theory of dislocations.

`kink_drag_coefficient` Specifies the (inverse) mobility of kink segments [Pa sec].

`kink_width` Width of a single kink in the screw direction [b].

`delta_H0` Prefactor ΔH_0 in formula for stress-dependent kink energy [eV].

Monte Carlo control parameters:

`num_events` Number of KP nucleation events to generate per screw segment and available kink direction.

`attempt_frequency` Rate prefactor for KP nucleation events [$\mathrm{sec}^{-1} b^{-1}$]