Alloy Application for SPPARKS

Introduction

The ideal audience for this project are people familiar with SPPARKS/LAMMPS or familiarity with Monte Carlo/molecular dynamics however this is not a requirement. The quick start guide will have you up and running in minutes. There are a few different example scripts you can use that range from easy to difficult and slow to fast computationally.

About SPPARKS

SPPARKS (Stochastic Parallel PARticle Kinetic Simulator) is a C/C++ application developed by Steve Plimpton at Sandia National Lab and released under the GPL license. The original source code and documentation can be found at https://spparks.sandia.gov. It serves as a scripting environment for physics based Monte Carlo simulations to run. Commands are placed in a file and loaded into the application using stdin.

About This Project

This project creates a new application in C/C++ similar to apps seen in the documentation of the command "app_style" for SPPARKS. "app_style alloy" is an on-lattice application that performs diffusion events with a probability related to the bonds between atomic pairs. Defining the probability this way is what allows us to simulate multiple types of atoms and observe phenomenon such as grain boundary formation. This is of particular importance to researchers as binary and ternary alloys have prominent usage in thermocouples, PV cells, semiconductors, metallic glasses and more.

The code that I have added to enable this new functionality is encapsulated in its own C++ class and can be found entirely in the files "app_alloy.cpp" and "app_alloy.h".The modularity of the code allows all the other app_styles to be used just as they are defined in their original documentation.

Alloy Application

"app_style alloy" is an on-lattice application that performs diffusion events with a probability of the Boltzmann factor p=e^(Q/kb*t). Where kb*t is the thermal energy of the system and Q is the sum of the bond energies between the site i and all of the neighbor atoms j. To use this application you must specify the bond energies using the bond_energy command detailed in this documentation.

Something to consider is what happens to atoms that diffuse to a lattice point with a coordination number of zero. There is no concept of a trajectory or velocity in this application so an incoming vapor of particles will not drive these particles back to the surface of the substrate. If allowed to diffuse in the simulation domain they will nucleate and cause crystals to grow some distance above the substrate. To alleviate this, atoms that have a coordination number of zero are considered desorbed atoms and are removed from the simulation domain.

Quick Start Guide

1. Open a Linux terminal.

2. Navigate to the "src" directory.

3. Type "make serial" to generate an executable called "spk_serial".

4. Run an example script by typing the command "./spk_serial<in_grain formation" (both files must be in the same directory).

5. View the images.

6. **(Optional)** Use the images to make a movie by typing "convert image*.ppm movie.gif".

Commands

General Commands

There are a wide array of commands that can be used for the input script. A generic application that uses some type of lattice structure to perform events usually has these commands to start with.

```
seed
app_style
dimension
lattice
region
create_box
create_sites
solve_style
```

This is by no means an exhaustive list. You should use "app_style alloy" to set the SPPARKS application to the one in this documentation. If you are not familiar with most of these commands or the general structure of a SPPARKS script, I suggest sticking to the format of one of the example scripts provided.

Unique to Appstyle Alloy

```
deposition rate dirx diry dirz d0 lo hi n
```

The deposition command allows atoms to deposit in competition with diffusion events. The parameters are:

parameter	description
rate	atoms deposited per second
dirx,diry,dirz	the direction of the incident vapor of atoms deposited
dtheta	allowable distance a deposited atom can be from original vapor target
lo,hi	max and min coordination number of deposition site
n	an integer defining the atomic species deposited

```
bond_energy i j q
```

The bond energy determines the strength q of an interaction between two atoms i and j. "bond_energy j i q" is equivalent to "bond_energy i j q" and defining the same atomic pair twice will throw an error. The parameters are as follows

parameter	description
i,j	an atomic pair defined by two integers
q	the strength of a single bond

Note: The integer defining atomic species is stored in the array labeled "i2" as seen in the "set" command. You can use the set command to define the type of atom and use "i1" to define whether a site is OCCUPIED or VACANT. If for example, you create a substrate using the set command you should set i1 and i2.

Input Scripts

Grain Formation

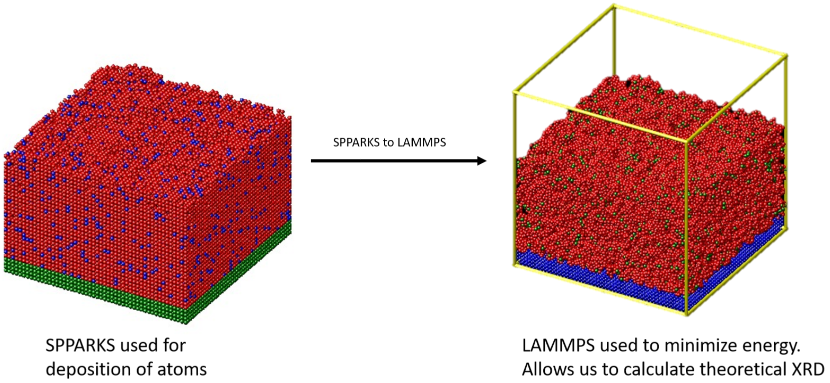
"in.grain_formation" is one of the examples included with the software in the scripts directory. It is one of the simpler simulations to run.Two atoms are co-deposited at 90 degrees to the substrate and the bond energies are defined in such a way that maximum segregation should occur. Running this script will produce a sequence of image files in the data folder. The filenames are padded with zeros so that it is easy to convert them to a GIF using ImageMagick which is a free software included with most linux systems.

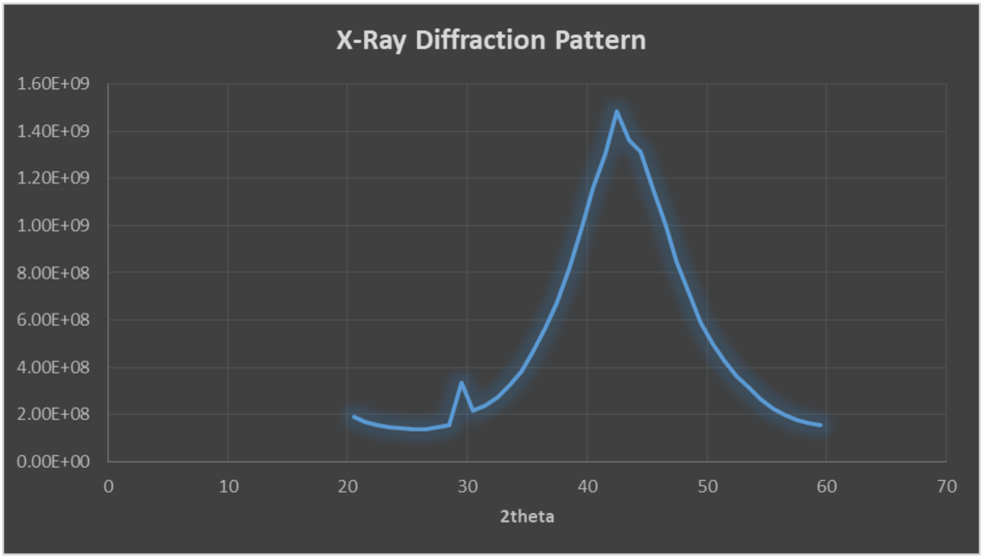
Oblique Angle Deposition

The file "in.oblique_angle" will perform co-deposition at angles larger than 85 degrees to the substrate. This causes a shadowing effect that produces nano-rods. As in the previous script the bond energies are defined to produce maximum segregation.

SPPARKS2LAMMPS

Simulation of 10:1 Ratio of Copper and Zirconium


SPPARKS to LAMMPS
SPPARKS used for deposition of atoms
LAMMPS used to minimize energy. Allows us to calculate theoretical XRD



This is one of the more advanced examples included in this repository that allows you to wrap your code between Monte Carlo simulations and molecular dynamic algorithims. It is a two part example that is contained in the folder titled "scripts/SPPARKS2LAMMPS". Run the script "in.spparks" using a SPPARKS executable. This will generate an image and a text file. The text file titled "sites.txt" serves to initialize LAMMPS with data about occupied sites. The "in.lammps" file should be ran with a lammps executable. The source code for LAMMP is not included here but can easily be download from Sandia's website. Building the code is similiar to SPPARKS with one exception being that there are a few user packages that you have the option to include in your build. For this example I have enabled the MANYBODY and USER-DIFFRACTION package. Executing "in.lammps" will generate an image after atom energies are minimized and relaxed. A theoretical xrd pattern will also be generated. The images do not take too long to produce but the XRD pattern can take a few hours at least.

Some General Comments

Compiling on Windows

It is possible to compile this software on Windows using Cygwin but this is highly discouraged. There are some concepts such as permission bits that won't translate well to Windows. If you do not have access to a Linux system it is easy to get started with one by using a virtual machine or imaging a usb with a copy of Linux.

VCU Thesis Readers

If you have stumbled upon this repository because you have read my thesis, please note that this software can perform the simulations but is not the exact format used at the time I wrote it. You can email me at marshalltc@vcu.edu for an exact copy if you would like.

Feedback

I am always looking for good feedback whether on this project or another. If you have any comments, suggestions, or have found a sofware bug please email me at marshalltc@vcu.edu.

Acknowledgements

Thank you to my adviser for helping me to create and use this software to perform simulations of binary alloys and publish my thesis at Virginia Commonwealth University.