Transactions of the Materials Research Society of Japan. Vol. 20

Flexible Architecture for Atomic Simulation on the Macintosh

Olof Hellman, Akira Seki and Shun-ichiro Tanaka

Tanaka SolidJunction Project, JRDC
1-1-1 Fukuura, Kanazawa-ku, Yokohama Japan

We demonstrate an application , kSan, which allows researchers easily to construct complex simulations of atomic structure. kSan works cooperatively with other Macintosh applications, so as to facilitate data input and data analysis. kSan is also extensible in that it supports a plug-in architecture for atomic potentials and data collection routines. We outline kSan's functionality and show examples of interaction with other applications.

1. INTRODUCTION

For a number of years now, off-the-shelf applications for PCs have implemented standards which make it easy for them to work together, for example, copy and paste. Additionally, some programs support sharing of functionality or other message passing standards. This means, for example, that a word-processor's spell checker can be implemented as a single piece of code, and any applications which are "spell checker-aware" are able to inherit the spell checker's ability. One benefit is that to make and use a different spell-checker, as is required for documents with multiple languages, the word processor need not be modified at all.

Scientific software benefits from a similar approach: by defining standards for different pieces of an application's functionality, code can be used by any number of applications, and the functions themselves may be easily modified. kSan takes a step in this direction by offering a fairly wide range of services for performing molecular dynamics and energy minimization of particle configurations. This means that researchers who wish to implement a new piece of the application (e.g. an interatomic potential), need not worry about file management, graphics display, communication with other applications, printing, etc., yet all these functions are available if the code is packaged so that kSan can use it.

2. THE APPLICATION

kSan performs traditional molecular dynamics calculations and atomic structure energy minimization, and serves as a viewer and editor for the ensemble of particles it works on. It provides standard functions such as file keeping, printing, and communication with other applications. Lastly, it serves as a manager for various plug-ins which extend its functionality.

The viewer allows researchers to see an ensemble of particles in perspective, and to manipulate the attributes of the view: magnification, perspective, depth shading, etc. It also allows viewing of subsets of the ensemble, which makes for easier viewing of complex structures: when viewing a grain boundary, a researcher may choose to only see atoms near the grain boundary. kSan can display many of the attributes of the ensemble as raw data, such as bond angles, particle energies, temperatures etc. in real-time as a simulation progresses. Particle colours can be keyed to attributes such as particle potential or particle kinetic energy, or to parameters such as group or type, allowing for more intuitive understanding of numerical data.

The editor allows researchers to change attributes of the simulation such as cell size and particle positions and velocities. kSan allows particles to be selected and grouped. In a simulation, constraints or forces may be applied to these groups, which permits the construction of complicated boundary

conditions. In this sense, the ensemble editor can be used to affect the attributes of a different ensemble -- the thermodynamic ensemble of free variables.

For example, one of the problems faced in the simulation of interfacial structure is the construction of a boundary condition on either side of the interface which approximates the bulk structure. One approach to this problem is to constrain a set of particles on either side of the boundary to remain in the zero-temperature bulk structure relative to each other, but to move freely as a group.[1] Using kSan, these sorts of constraints are applied by point and click. Many different types of constraints may be applied: thermal time constant, equilibrium temperature, motion constraints, and applied force may all be independently manipulated for each group.

The calculation engine performs an iterative calculation based on either Gear [2] or Verlet [3] molecular dynamics algorithms, or performs energy minimization by a conjugate gradient method. Parameters for these algorithms may be manipulated by the researcher.

kSan supports export of graphics and data via copy and paste so that researchers can use standard graphing or illustration programs to analyze, plot, print and transform data. kSan also supports the Macintosh interapplication communication system via AppleEvents, which permits kSan to be controlled via a script, either to extract automatically data during runtime, or to manipulate attributes of the ensemble itself as a simulation proceeds. Data extracted during runtime in this manner may be passed to another AppleEvent-aware program, allowing automated production of graphs, animations, etc.

kSan requires a Power Macintosh. As an additional bonus for Japanese users, the user interface runs completely in Japanese (except for those pesky error messages) if the Japanese script is enabled, including vocabulary for the Japanese AppleScript dialect. kSan is available at the Materials Research Society ftp archives, or through online software libraries such as info-mac and MacSciTech.

3. EXTENSIBILITY.

Much of kSan's flexibility comes from the fact that important functions are implemented as plug-in modules: these pieces of code are dynamically linked at runtime, so that they can be swapped in and out without modifying the main part of the application. The two essential pieces of the simulation which are plug-ins are the interatomic potential and interatomic screening function. Only one potential and one screening function are active at any given time. The advantages for researchers developing interatomic potentials should be obvious: one need only write and compile code defining the potential, and the rest of the functionality of kSan comes for free.

The potentials implemented with this version of kSan include Lennard-Jones [4], Tersoff's silicon potential [5], Tersoff's SiGeC [6] potential and Baskes' Modified Embedded Atom Potential (MEAM) for Fe, Cu, Cr, Ag, etc.[7] kSan performs accounting of nearest neighbors and atom types, so if a potential can benefit from a short range cutoff, or if the potential supports multiple atom types, these potentials can be implemented with very little work.

The screening functionplug-in allows potentials which use short range cutoffs to use different cutoff functions. The choice of screening function is often crucial for accurately modelling a system. The parameters used in an interatomic potential such as the MEAM are often fit to apply to the bulk structure and a small number of simple defect structures assuming a radially symmetric cutoff. However, in performing simulations involving more complicated defect structures such as grain boundaries, a radially symmetric cutoff might not be as useful.

Data collection and analysis is provided by a third type of plug-in module. Any number of these plug-ins may be active at any time. Each of the active data collection plug-ins is called after a set number of iterations, allowing periodic collection and display of data.

The data collection plug-ins implemented at this time are Thermometer, Radial Distribution Function and Barometer. Almost any type of physical property measurement could be implemented as a plug-in and

```
tell application "kSan"
  set hcpA to 2.56 -- nearest neighbor distance in Cu
  set hcpB to 1.732 * hcpA -- root three
  set hcpC to 1.633 * hcpA -- this is two times the close-packed planar spacing
  set xCells to 4.0 -- make 4 x unit cells in each closepacked layer
  set yCells to 3.0 -- make 3 y unit cells in each closepacked layer
  set zCells to 3.0 -- make 3 ab layers
  set cell x dimension to hcpA * xCells
  set cell y dimension to hcpB * yCells
  set cell z dimension to hcpC * (zCells + 0.5) -- +0.5 for the extra c layer
  repeat with zValue from -1 to zCells - 2 -- make an ab layer each pass through loop
    set zCoord to zValue * hcpC
    repeat with yValue from -1 to yCells - 2
      set yCoord to yValue * hcpB
      repeat with xValue from -1 to xCells - 2
        set xCoord to xValue * hcpA
        create particle at {xCoord, yCoord, zCoord}
        create particle at {xCoord+(hcpA/2), yCoord+(hcpB/2), zCoord}
        create particle at {xCoord+(hcpA/2), yCoord+(hcpB/6), zCoord+(hcpC/2)}
        create particle at {xCoord, yCoord+(hcpB * 2/3), zCoord+(hcpC/2)}
          -- the first two are in the a layer, the last two are in the b layer
      end repeat -- go to next x unit cell
    end repeat -- go to next y unit cell
  end repeat -- go to next ab layer
  - - finished ababab structure. Now, add a c layer, making an abcab stacking
  set zCoord to (zCells - 1) * hcpC
  repeat with yValue from -1 to yCells - 2
    set yCoord to yValue * hcpB
    repeat with xValue from -1 to xCells - 2
      set xCoord to xValue * hcpA
      create particle at {xCoord, yCoord + (hcpB/3), zCoord}
      create particle at {xCoord + (hcpA/2), yCoord+(hcpB * 5/6), zCoord}
    end repeat
  end repeat
end tell
```

Listing 1. This script creates an ensemble of particles with a hybrid fcc/hcp stacking

used with any of the potentials or iteration methods. We plan to implement more plug-ins for calculating and displaying the Boltzmann H function, the internal virial, the phonon spectrum, etc.

Data collection plug-ins can provide real-time visual feedback for the researcher. For example, Radial Distribution Function may be set to calculate the RDF of the ensemble for each iteration and display it in real time. These plug-ins might also be used to periodically collect some set of data and store it on disk. Using the AppleEvent mechanism, these plug-ins can send messages to kSan to change paramters of the simulation, thereby changing the simulation

```
tell application "kSan"
    select (every particle whose potential is greater than -3.2)
    create new window with selections
end tell
```

Listing 2. This script makes a new window containing only the particles with a certain potential energy

as a function of some calculated parameter. For example, a virial plug-in might change the size of the unit cell in response to the measured stress tensor.

4. SCRIPTING EXAMPLES

The plug-in architecture of data collection and interatomic potentials are examples of *extensibility*. Another dimension of flexibility comes via *interoperability*, the ability to communicate with other applications. We demonstrate interoperability with two examples of scripts written in the AppleScript language. In each case, these are operations which could be done by hand, by point and click. However, because kSan can accept and respond to commands from external applications like the script editor, very tedious tasks can be made rather simple.

For example, a researcher might use AppleScript to define a initial configuration for particles in the cell. Listing 1 is a script which creates a cell containing particles in an hcp structure (ababab). Then, it inserts one layer in fcc stacking (c), thereby creating a hybrid hcp/fcc structure (ababcab). Because AppleScript is a full featured programming language, arbitrarily complex structures may be created in a similar fashion.

As a second example, we demonstrate some of the graphics capabilities of scripting kSan. Some researchers might have a rather complex ensemble and wish to use the multiple windowing environment in kSan to view ensemble data in a more intuitive way. Listing 2 is a script which selects all of the particles with a potential energy greater than a certain limit, and makes a new window which contains only those selections.

Note how complicated an action is performed by this small script. Because of AppleScript's ability to interpret subordinate clauses like "**whose** potential **is greater than** -3.2", a single statement performs an action that would have required a loop and an if statement in a more traditional language. Also note how easy it might be for a non-programmer to understand the action performed by the script.

5. CONCLUSIONS

kSan is a flexible application for performing simulations of atomic structure. Because of its point and click user interface, we expect that researchers will find it easy to construct and begin simulations. Because of its extensibility, we expect a number of researchers to use and extend its functionality to a number of different types of calcultions. Because of its interoperability, we expect researchers to use kSan cooperatively with other Macintosh applications, including scripting environments.

6. REFERENCES

1. J..F. Lutsko, D. Wolf, S. Yip, S.R. Phillpot and T. Nguyen, Phs. Rev. B38 11572 (1988).
2. J.M. Haile, Molecular Dynamics Simulation: Elementary Methods, J. Wiley & Sons, New York (1992). ISBN 0-471-81966-2
3. L. Verlet, Phys. Rev. 159 98 (1967).
4. J.E.Lennard-Jones, Proc. Roy. Soc. (Lond.), 106A 441 (1924).
5. J. Tersoff, Phys. Rev. B37 (12) 6991 (1988). A
6. J. Tersoff, Phys. Rev. B38 (14) 9902 (1988). B
7. M.I. Baskes, Phys. Rev. B46 (5) 2727 (1992). That last M in MEAM stands for Method, not for Potential. Just in case you were wondering.