THERMAL PROPERTIES OF YTTRIUM ALUMINUM GARNETT FROM MOLECULAR DYNAMICS SIMULATIONS

by

Majid S. alDosari

Thesis

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

in

Mechanical Engineering

May, 2012

Nashville, Tennessee

APPROVED:

Professor Greg Walker, Chair

Professor Deyu Li

Professor Jason Valentine

AKNOWLEDGEMENTS

Acknowledgment is mainly to Saudi ARAMCO and its Advanced Degree Program which sponsored my degree. Also, every Thermal Engineering Lab member has contributed in some way to this work. But, it is difficult to order their contribution so they are listed here according to their seating in counter-clockwise order starting from the lab's only window: Nicholas Roberts, Terry Musho, Ebonee Walker, Mike Myers, and Rachel Hansel. In addition, I'd like to acknowledge the Advanced Computing Center for Research and Engineering (ACCRE) at Vanderbilt for providing the necessary computational resources. Last but not least, I appreciate Dr. Walker's direction in this work which I have found to be interesting, enjoyable, and satisfying.

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

A	two-body potential parameter
A_c	cross-sectional area
B	two-body potential parameter
E	total energy
E_{ts}	change in system energy due to a thermostat
E_{ts}^{avg}	average of the absolute values of E_{ts}
MD	molecular dynamics
N_{pairs}	number of pairs
NPT	isothermal and isobaric ensemble: conserves moles (N), pressure (P) , and temperature (T)
NVE	microcanonical ensemble: conserves moles, volume (V), and energy (E)
NVT	canonical ensemble: conserves moles, volume, and temperature
RDF	radial distribution function
T	temperature
U	potential energy
V	volume
V_0	initial volume
a	lattice spacing
c_p	constant-pressure specific heat
c_v	constant-volume specific heat
f_{ij}	force of atom i on atom j
g	radial distribution function
k	thermal conductivity
k_B	Boltzmann's constant
p	average number of atoms pairs
q	elementary charge
$\dot{q}$	heat flux
r_{ij}	distance between atoms i and j
$r_{0,ij}$	atomic bonding cutoff between atoms i and j
t	time
u.c.	unit cell
v	velocity
x	fraction of Al randomly substituted with Ga: $Y_3(Al_{1\text{-}x}Ga_x)_5O_{12}$
Λ	mean free path
Ω	three-body angular expression
α	thermal expansion
γ	three-body potential parameter energy
δ	Lindemann index
λ	three-body potential parameter
θ_{jik}	angle $\widehat{jik}$
θ_0	three-body potential parameter

CHAPTER I

INTRODUCTION

Yttrium Aluminum Garnet (YAG, $Y_3Al_5O_{12}$) is a synthetic crystal with a complicated atomic structure. Doped varieties are used for solid state lighting [3], as thermographic phosphors [4], in thermal barriers [5], and as lasing mediums [6]. The inherent thermal design issues of these applications motivate the study of the thermal transport and bulk properties in YAG. However, recent efforts to exploit and understand the small-scale behavior of YAG in, for example, radiation environments and luminescent excitation has prompted the study of the material from a atomistic point of view. In these detailed research studies quantities such as dominant vibrational modes, density of states and other non-equilibrium descriptions are important to deducing the interactions of thermal quantities with other physical effects.

In the foregoing work, a molecular dynamics simulation is used to extract bulk thermal properties of the YAG structure. By comparing the results to established bulk values, we can learn about the efficacy of the simulation approach to model thermal properties and transport of pure YAG crystals. This effort will lay the ground work for molecular dynamics in further studies of more detailed properties that can not be compared directly with measurements.

Since YAG has a relatively low thermal conductivity, its thermal transport is primarily governed by phonons, which are inferred from atomic motion. Therefore, classical molecular dynamics (MD) simulations can offer insight into the thermal properties of YAG of varying system parameters as long as temperatures are above 50 K, approximately 10% of the Debye temperature.

In the work introduced by Hansel [4], gallium is substituted for aluminum in various percentages to study its effect on luminescence. In this work, the perfect YAG structure will be defects by substituting aluminum with gallium in varying percentages. Subsequent calculation of the thermal properties will allow us to study the substitutional effects on constant-pressure specific heat, lattice constant, thermal expansion, and thermal conductivity. Realize that for these simulations, the mass of the aluminum atom will be changed to that of gallium, but the interatomic potential will remain the same as the aluminum. Therefore, we should be able to determine whether the mass of the substitution or the bonding dominates the thermal properties by examining the deviation of the simulation from expected values.

In addition to the thermal properties of substituted YAG, we will consider the melting temperature of pure YAG and extrapolation of the thermal conductivity to infinite systems for additional validation. However, molecular dynamics is notoriously poor at predicting melting temperature because melting is a process that requires localized initiation [7]. In periodic crystals used in simulations, the localized initiation does not exist; therefore, simulated melting temperatures are usually far greater than measured melting temperatures. Consequently, we will attempt to recover the melting temperature by introducing vacancies to mimic real crystals. To deduce the macroscale thermal conductivity, the system size will be systematically varied to provide adequate data for extrapolation. We also know that the extrapolation of size dependent data can be imprecise for complex systems, so we want to evaluate the efficacy of using standard extrapolation techniques to recover bulk thermal conductivity.

Thermal properties of pure, crystalline YAG are reported in Table 1 for comparison to results from our simulations.

Table 1 THERMAL PROPERTIES CRYSTALLINE OF YAG

Melting Temperature, T_m	2213 K [8]
Thermal Expansion, α	$7.0 \cdot 10^{-6}$ K^{-1} @300 K [9]
Specific Heat, c_p	0.6 $J \cdot g^{-1} K^{-1}$ @300 K [10] [11]
Thermal Conductivity, k	10–14 $W \cdot m^{-1} K^{-1}$ [12]
Debye Temperature, T_D	750 K [13]

CHAPTER II

YAG MODEL

YAG crystallizes in a highly symmetric cubic structure (detailed in Reference [2]). The conventional unit cell of YAG contains 160 atoms comprising of 24 yttrium, 40 aluminum, and 96 oxygen atoms; with a lattice parameter of 12.01 Å [14]. The relative arrangement of Y, O, and Al can be described with polyhedra as shown in Figure 2, while the symmetry can be seen in Figure 1.

Jun, et al. [15] have modeled the complex structure of pure YAG for classical molecular dynamics simulation by fitting elasticity and lattice constant data to representative potential energy functions. We are leveraging this work to subsequently calculate thermal properties from these potentials that were designed for mechanical properties. As such, non-transport properties such as thermal expansion and specific heat should compare well. However, thermal conductivity calculations, which require precise description of the anharmonicty may deviate from an insufficiently tuned interatomic potential. Therefore the focus is to evaluate the use of the already generated potentials for thermal (both transport and non-transport) properties. Moreover, we will substitute aluminum for gallium by changing only the mass of the oscillators; the interatomic potential will remain the same for the substitutional studies.

The potential energy for any pair of atoms (ions), i and j, are given by the sum of a Coulombic and exponential term given by

$$U_{ij} = c\frac{q_i q_j}{r_{ij}} + A_{ij}\exp\left(-\frac{r_{ij}}{B_{ij}}\right) \tag{1}$$

where c is an energy conversion constant equal to 14.4 eV·Å $\cdot\, e^{-1}$.

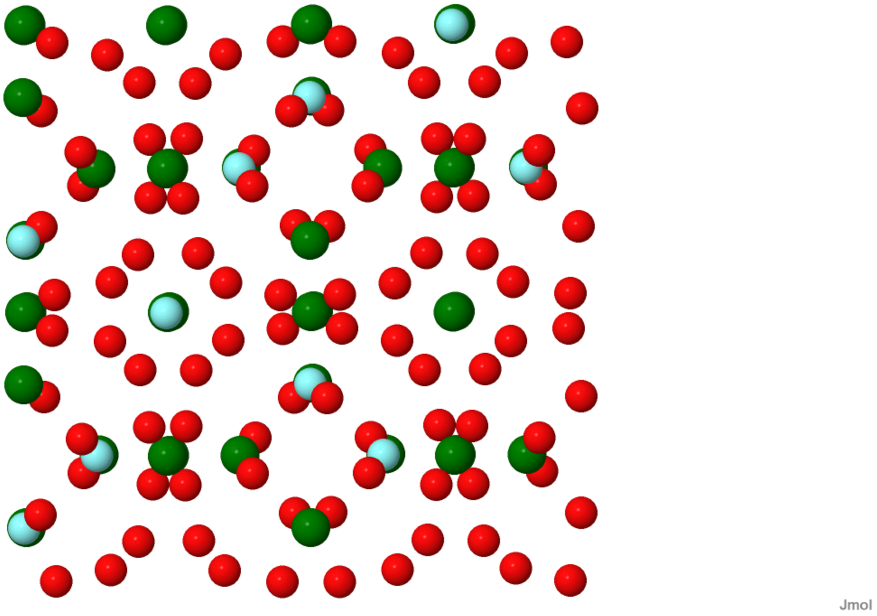

Figure 1 CONVENTIONAL UNIT CELL YAG SYMMETRY ON [1 0 0] FACE. Red:O, Green:Al, Blue:Y. (Image generated by Jmol [1])

Table 2 TWO-BODY PARAMETERS

ij	A_{ij}(eV)	B_{ij}^{-1}(Å^{-1})	q_i (e)
O-O	2449.44	3.44	-2
Al-O	1740.31	3.44	$+3$
Y-O	1250.85	2.86	$+3$
Al-Al	312.11	14.06	
Y-Y	245.14	14.06	
Y-Al	256.55	14.06	

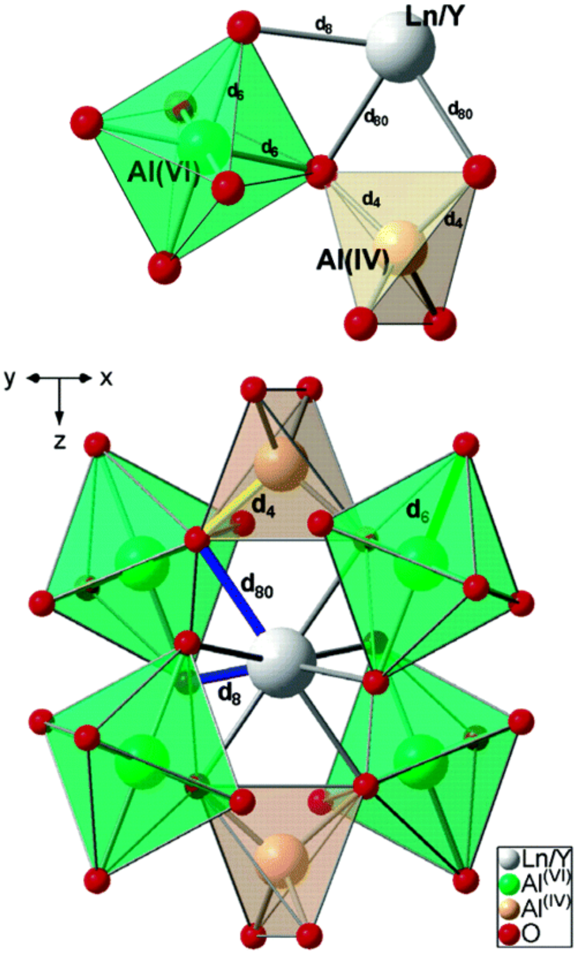

Figure 2 POLYHEDRA IN YAG [2] (with permission from the American Chemical Society)

Table 3 THREE-BODY PARAMETERS

jik	λ_{jik} (eV)	γ (Å)	$r_{0,ij}$ (Å)	$\theta_{0,jik}$ (°)
Al/Y–O–Al/Y	6.242	2.0	2.6	109.5
O–Al–O	149.324	2.8	3.0	109.5
O–Y–O	168.250	2.8	3.0	109.5

In addition to pair potentials, three-body potentials were employed to enforce angular bonding between certain triplets of atoms as seen in Figure 2. For a triplet of atoms; i, j, and k; the three-body interaction, U_{jik}, is expressed as

$$U_{jik} = \begin{cases} \lambda_{jik} \exp\left(\frac{\gamma_{ij}}{r_{ij}-r_{ij,0}} + \frac{\gamma_{ik}}{r_{ik}-r_{ik,0}}\right)\Omega_{jik} & r_{ij} < r_{ij,0} \text{ and } r_{ik} < r_{ik,0} \\ 0 & r_{ij} \geq r_{ij,0} \text{ or } r_{ik} \geq r_{ik,0} \end{cases}. \tag{2}$$

Ω_{jik} is an angular factor whose functional form depends on which triplet of atoms is being considered. For bonds where j and k represent oxygen atoms, and i represents an aluminum or yttrium atom (O–Al/Y–O), Ω_{jik} is expressed as

$$\Omega_{jik} = [(\cos\theta_{jik} - \cos\theta^0_{jik})\sin\theta_{jik}\cos\theta_{jik}]^2. \tag{3}$$

For triplets Al/Y–O–Al/Y, Ω_{jik} is

$$\Omega_{jik} = (\cos\theta_{jik} - \cos\theta^0_{jik})^2. \tag{4}$$

The parameters of Equations 2, 3, and 4 are in Table 3.

All simulations were performed using the June 8, 2011 version of the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) package [16]. Furthermore, all simulations used the Verlet algorithm [17] for time integration. Periodic boundary conditions were imposed in all directions on all simulations to balance the repulsive forces from similarly charged two-body interactions. Periodic boundary conditions are also required to compute Coulombic interactions in reciprocal space.

The two-body interactions expressed by Equation 1 are implemented using the included Born-Mayer-Huggins potential [18] where Coulombic interactions greater than a specified cutoff were computed in k-space using a particle-particle particle-mesh (PPPM) solver. The LAMMPS tool, which does not include the proposed potential inherently, was modified to include the potential energies expressed by Equations 2, 3, and 4 (see Appendix).

LAMMPS SOURCE MODIFICATION TEST

The correctness of the source modification was tested by checking energy conservation in an NVE ensemble for one million timesteps (Figures II and II). A mistake in the code should not conserve energy at all, while an improper simulation parameter should result in poor energy conservation. The simulations ran with a 10^{-4} PPPM precision, a 4 Å cutoff for non-Coulombic two-body interactions, and a 7 Å cutoff for real space Coulombic interactions, and a .002 ps timestep. These parameters were found to conserve energy while maintaining decent execution speed. The simulation parameters used for the results presented here were the same or more stringent.

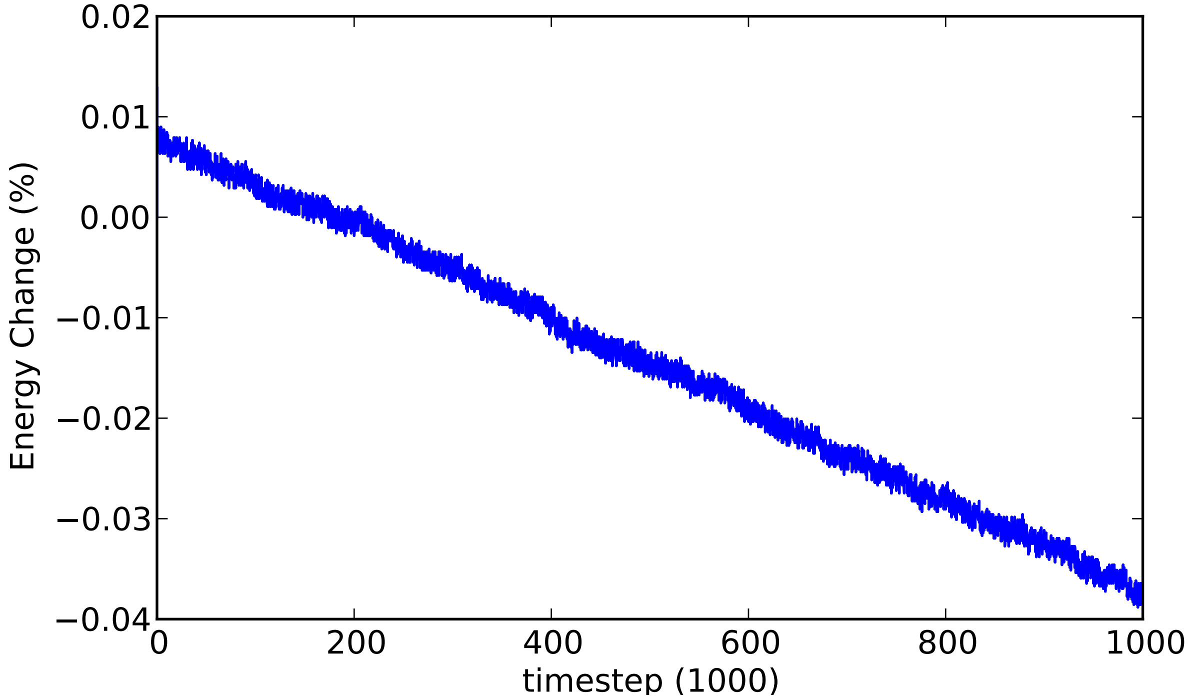

Figure 3 ENERGY CONSERVATION OF 2×2×2 UNIT CELLS IN NVE ENSEMBLE FOR x=0

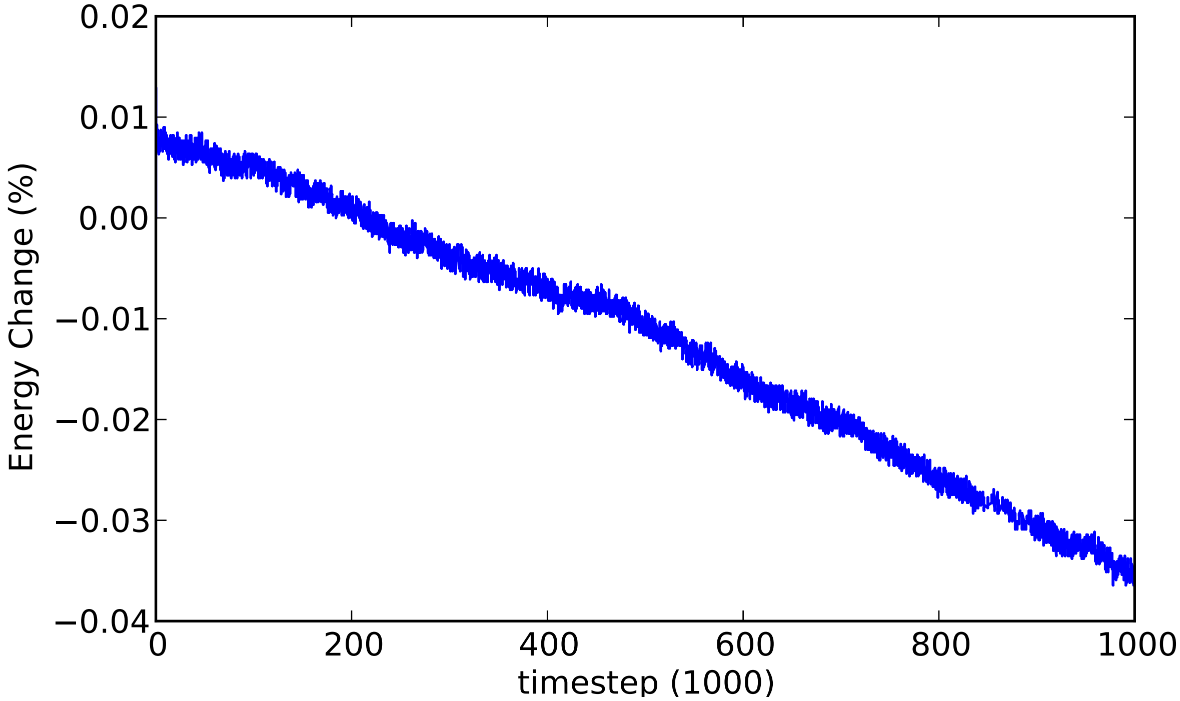

Figure 4 ENERGY CONSERVATION OF 2×2×2 UNIT CELLS IN NVE ENSEMBLE FOR *x*=50 %

CHAPTER III

LATTICE CONSTANT, SPECIFIC HEAT CAPACITY, THERMAL EXPANSION

This section describes how the constant-pressure specific heat, thermal expansion, and the lattice constant are found from simulated heating. The agreement of these quantities with experimental values indicates whether the interatomic potential is suitable for non-transport thermal properties.

3.1 Simulation Setup

a, c_p, and α are calculated from five simulations to represent a random aluminum substitution percentage of 0, 25, 50, 75, and 100 percent. The system size was 2×2×2 unit cells (1280 atoms) for all simulations. Each simulation was initialized by selecting a normally distributed random velocity to give a temperature of 10 K. Then, using NPT time integration, the system is thermostatted at 10 K and barostatted at 10 bars for 80,000 timesteps. Subsequently, the temperature is ramped up by 100 K. After that, the system is again barostatted at 10 bars, thermostatted at the new temperature of 110 K, and time-integrated for another 80,000 timesteps. This process is repeated until the system reaches a temperature of 3000 K. Except for aluminum substitution, this simulation setup closely follows that of Jun [15] to allow for comparison.

The simulations ran with a 0.002 ps timestep, 10^{-4} PPPM precision, and a 7 Å cutoff (for two-body interactions). The damping parameter on the thermostat was 1 ps while the damping parameter on the barostat was 10 ps. An additional drag factor of 1 was imposed to dampen unstable oscillations.

A time series for the total energy, temperature, pressure, and the lattice constant was collected from the simulation by a time averaging procedure: the quantities are sampled

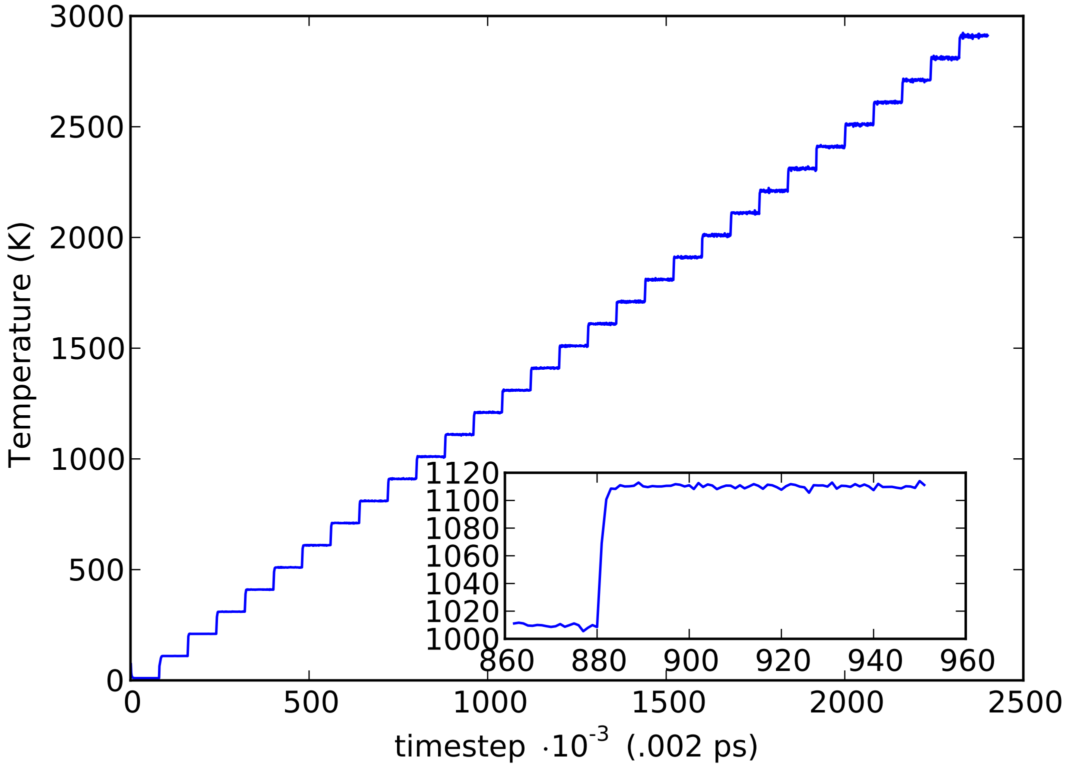

Figure 5 TEMPERATURE VS. TIME FOR 25% Al SUBSTITUTION

every 10 timesteps 100 times, and subsequently averaged. This average is then recorded.

3.2 Results and Discussion

The stability of kinetic energy in the simulation can be seen in the time series in Figure 5 where the temperature is increased every 80,000 time steps. Variations in each temperature "step" were within 10 K as seen in the inset of Figure 5 while exhibiting almost no long-term transients over the 80,000 times steps. Therefore, we conclude that 80,000 time steps is sufficient to produce an equilibrated system for each temperature. Similarly, the lattice parameter, *a*, exhibited variations of 0.1% at most for each temperature.

From these data, the relationship between temperature and total energy can be estimated, and it was found to be linear across the whole range of temperatures. Therefore, the specific heat, which is proportional to the slope of the energy/temperature relationship is constant in temperature.

$$c_p = \frac{\Delta E}{m\Delta T}. \tag{5}$$

In a classical simulation, this finding is expected and the value should be similar to that of the law of Dulong and Petit ($c_p = 3Nk_B$, where N is the number of oscillators). If we assume the density of YAG is $\rho = 4.56\,\mathrm{g/cm^3}$ and the lattice constant is $a = 12.01\,\text{Å}$, then the Dulong and Petit specific heat reduces to

$$c_p = 3\frac{160\,\text{atoms/unit cell}}{12.01\,\text{Å/unit cell}}\frac{k_B}{\rho} = 0.84\,\frac{\mathrm{J}}{\mathrm{g\,K}} \tag{6}$$

Figure 6 shows c_p as a function of the percentage substitution. Note that the law of Dulong and Petit compares well with 0% substitution. However, the calculated c_p of pure YAG is about 40% greater than the experimental value of $0.6\,\mathrm{J/g\,K}$. The discrepancy is probably due to imperfections in the measured materials. We do not have density information for the gallium substituted varieties, so direct comparison for the substituted varieties is not possible. Because $\Delta E/\Delta T$ were close to equal for all simulations, c_p is primarily a function of mass as indicated by the trend for substituted systems. The molecular weight of YGG (100% substitution) compared to YAG is 1.36 times larger. Therefore, the specific heat should be $1/1.36 = 0.74$ times smaller, or about $0.64\,\mathrm{J/kg\,K}$. This result is exactly the result of the molecular dynamics calculation.

A linear relationship was also found between temperature and lattice constant for all simulations. Therefore, the coefficient of thermal expansion α, which is proportional to the slope of this relationship, is a constant.

$$\alpha = \frac{\Delta V}{V_0 \cdot 3\Delta T}. \tag{7}$$

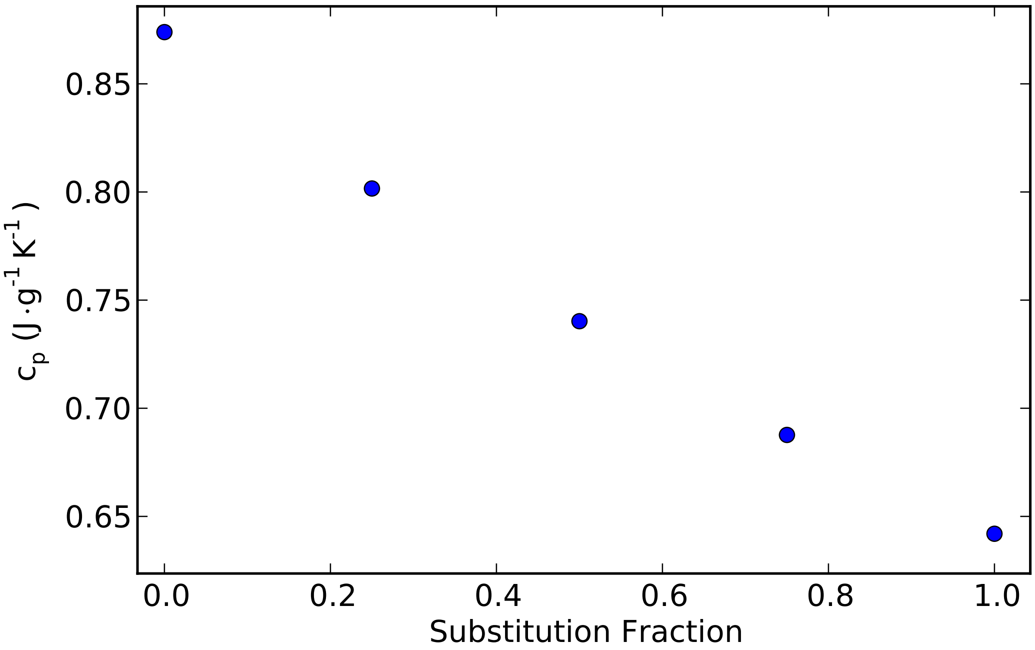

Figure 6 c_p VS. SUBSTITUTION FRACTION, x

The resulting thermal expansion of $\alpha = 7.8 \cdot 10^{-6}\,\mathrm{K}^{-1}$ for all simulations showed no dependence on aluminum substitution. This is expected given that the potential energy functions were the same for the substituted atoms. Nevertheless, this compares well with the established figure of $7.0 \times 10^{-6}\,\mathrm{K}^{-1}$ for pure YAG. As a result, a fit of the lattice constant data as a function of temperature yields

$$a = 9.67 \cdot 10^{-5} T \left(\frac{\text{Å}}{\text{K}}\right) + 12.4\,(\text{Å})\,. \tag{8}$$

So at 300 K, the lattice constant is 12.43 Å which is 3.6% greater than published figures [14] but only 0.6% greater than the result from MD simulations of Reference [15]. However, we can not validate the linear thermal expansion coefficient for gallium gallium-substituted YAG, and these values should be questioned.

CHAPTER IV

MELTING TEMPERATURE

The melting temperature can inferred from three independent measures: 1) system potential energy, 2) the radial distribution function (RDF), and 3) the Lindemann index δ [19]. All of these metrics indicate that atoms have changed their local environment and all are a function of temperature.

The RDF is defined as

$$g(r) = \lim_{dr \to 0} \frac{p(r)}{4\pi \left(N_{pairs}/V\right) r^2 dr}. \qquad (9)$$

where r is the distance between a pair of particles, $p(r)$ is the average number of atom pairs found at a distance between r and $r + dr$, V is the volume of the system, and N_{pairs} is the number of unique pairs of atoms. The RDF can be interpreted as the probability of finding an atom at a distance r relative to that of an ideal gas. In the solid phase, RDFs are characterized by peaks which become less pronounced with increasing temperature [20] indicating less structural order. Therefore, as the substance melts, the distance between atomic sites becomes blurred.

The system-averaged Lindemann index is an average over unique atom pairs as expressed in Equation 10.

$$\delta = \frac{2}{N(N-1)} \sum_{i,j(i \neq j)}^{N} \frac{\sqrt{\left\langle \mathbf{r}_{ij}^2 \right\rangle - \left\langle \mathbf{r}_{ij} \right\rangle^2}}{\left\langle \mathbf{r}_{ij} \right\rangle} \qquad (10)$$

It measures disorder and the mobility of the atoms relative the distances between them. During a heating process, the index increases linearly with temperature. Deviation from linearity is taken as a change in the structure.

Finally, changes in the trend of potential energy as a function of temperature also indicates structural changes.

Simulation Setup

The simulations devised to predict melting temperature are based on the simulations that produced data for lattice constant, specific heat and thermal expansion. However, because periodic boundaries are used to mimic an infinite crystal, there are no surfaces to initiate melting. As described previously, this lack of defect structure delays the onset of melting to physically unrealistic values. Therefore, vacancies were introduced by removing 6 random aluminum atoms in a $2\times2\times2$ unit cell system. To study the effects of number fo defects on the melting process, another simulation of the same size was conducted with 15 randomly selected aluminum atoms removed. In both simulations, the heating process started from 1500 K and proceeded to 3500 K in 10 K increments. Each temperature was kept for 100,000 time steps.

The potential energy, temperature, pressure, and the lattice constant were collected in time from the simulation by the same averaging procedure as before. The time step, potential cutoff, PPPM precision, and thermostatting were also the same. Meanwhile, the RDF was calculated every 10 timesteps. To calculate the Lindemann index, the coordinates for 20 randomly chosen atoms were sampled every 5 timesteps. In a separate simulation it was found that the summands in the Lindemann index follow a well-defined distribution so that the Lindemann index evaluated with only 20 atoms was within 2% of the evaluation using all atoms. Sampling the atom trajectories in this way saves significant computational resources.

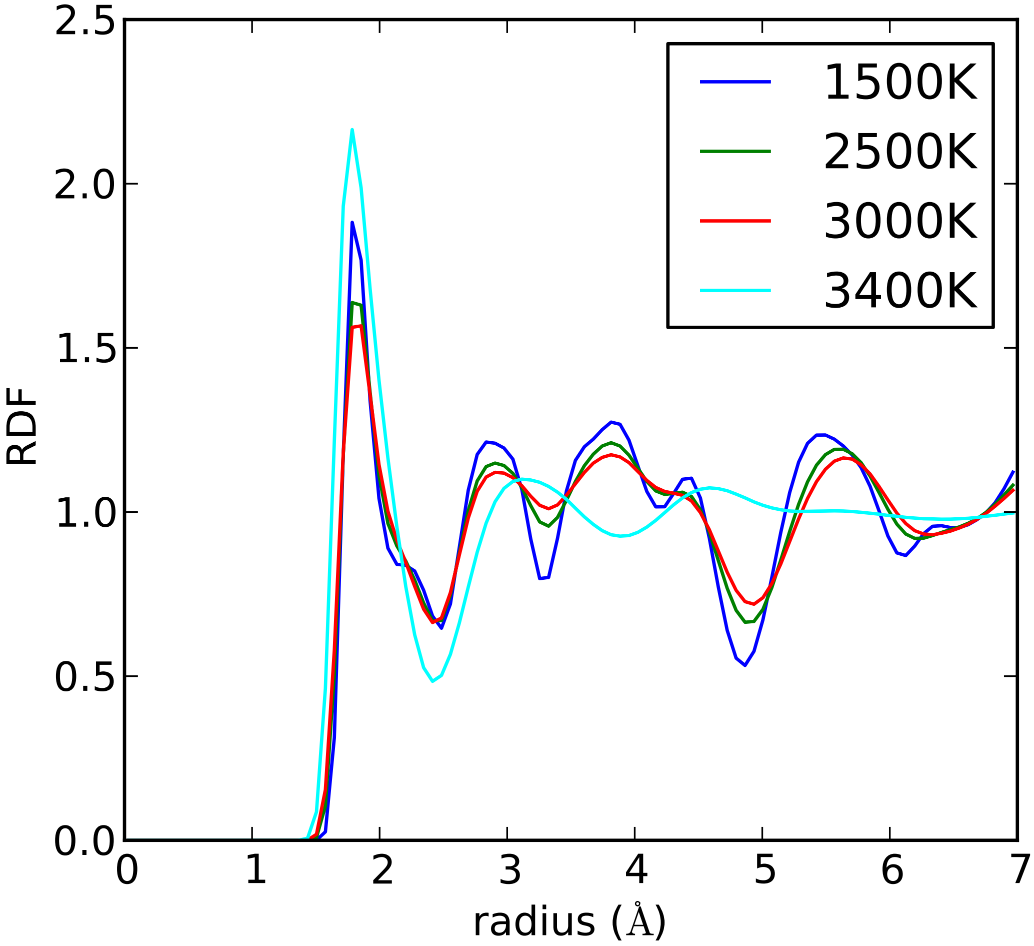

Figure 7 RADIAL DISTRIBUTION FUNCTION FOR 6 EVACUATED ALUMINUM ATOMS

4.1 Results and Discussion

The RDF of both simulations were similar at the temperatures plotted in Figure 7. We see a systematic but small loss of order between 1500K and 3000K. However, the difference in the RDF between 300K and 3400K is dramatic indicating that atoms are staring to migrate within the lattice and that the melting temperature is between these values. This transition indicates a loss of order beyond 4Å.

The potential energy also shows a dramatic change between the temperatures iden-

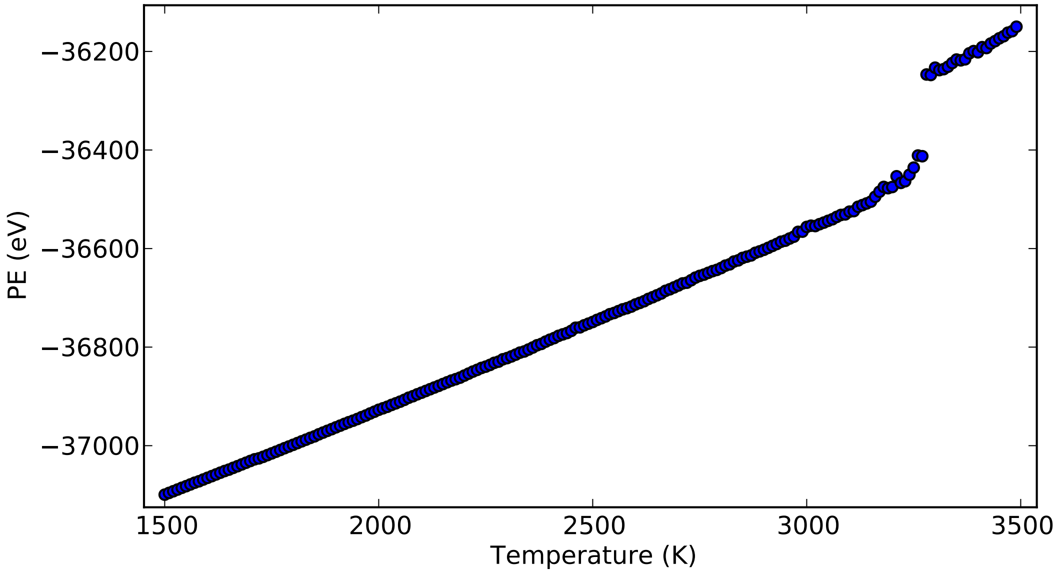

Figure 8 POTENTIAL ENERGY FOR 6 EVACUATED ALUMINUM ATOMS

tified by the RDF. An abrupt transition is seen by looking at a plot of potential energy versus temperature [21] as in Figures 8 and 9. Once the atoms begin to migrate relative to each other, which is the onset of melting, the potential energy remains large as the atoms are not constrained to the bottom of the potential well formed between atoms. The transitions occurred at two temperatures: one at about 3075 K and the other at approximately 3250 K. As expected the more highly defected material melts sooner than the less defected material. These results corroborate those found from the RDF analysis.

Meanwhile, the Lindemann index for the systems, shown in Figures 10 and 11, shows transitions occurring at temperatures below 3000 K by deviating from linearity. However the transitions were hundreds of degrees above the experimental melting temperature of YAG (2213 K). Nonetheless, the deviation from linearity is due to the coexistence of the liquid and solid state in the material (dynamic coexistence) [21] where the YAG ceases to be a crystal and is instead broken down into mobile clusters. These clusters must have all elements (Y, Al, and O) in order to be cohesive since the force between like atoms

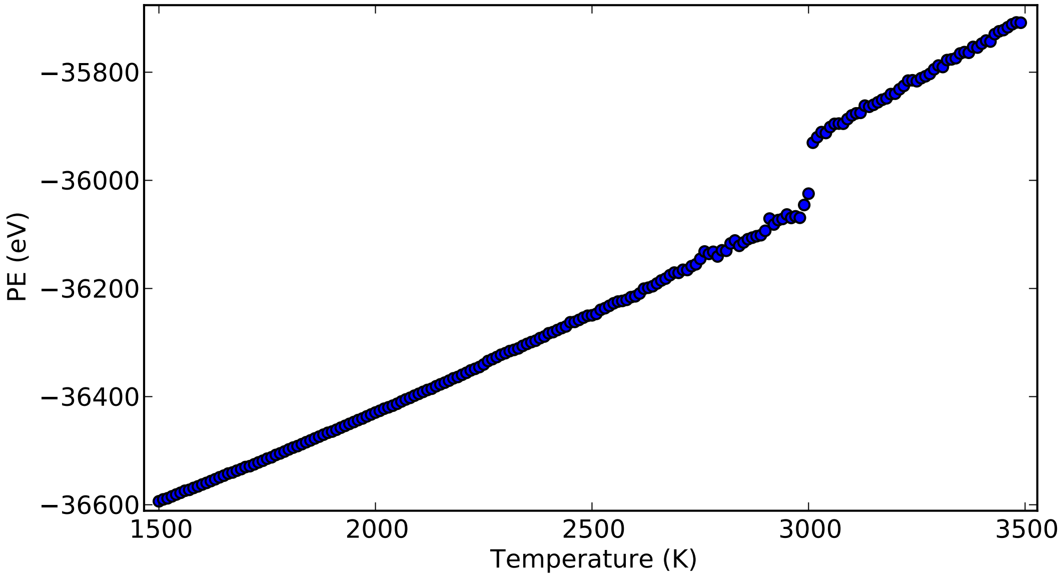

Figure 9 POTENTIAL ENERGY FOR 15 EVACUATED ALUMINUM ATOMS

is repulsive. So, oxygen may be seen as a binder of yttrium and aluminum. This explains why the RDF does not show much change through most of the temperature range. The RDF captures the relatively short-range order of these clusters while they may be mobile in the simulation. The defectiveness of the structure is more apparent with the more defective system as seen Figure 11 in comparison to Figure 10. The system with more vacancies is less stable. So, precisely determining the melting temperature from these simulations is difficult given this transition region over a wide temperature range. Furthermore, the configurations may be stable at a certain temperature in the timescale of the simulation but not in the timescale required for melting [7]. Even further, once the YAG structure is broken, the potentials used are no longer strictly applicable.

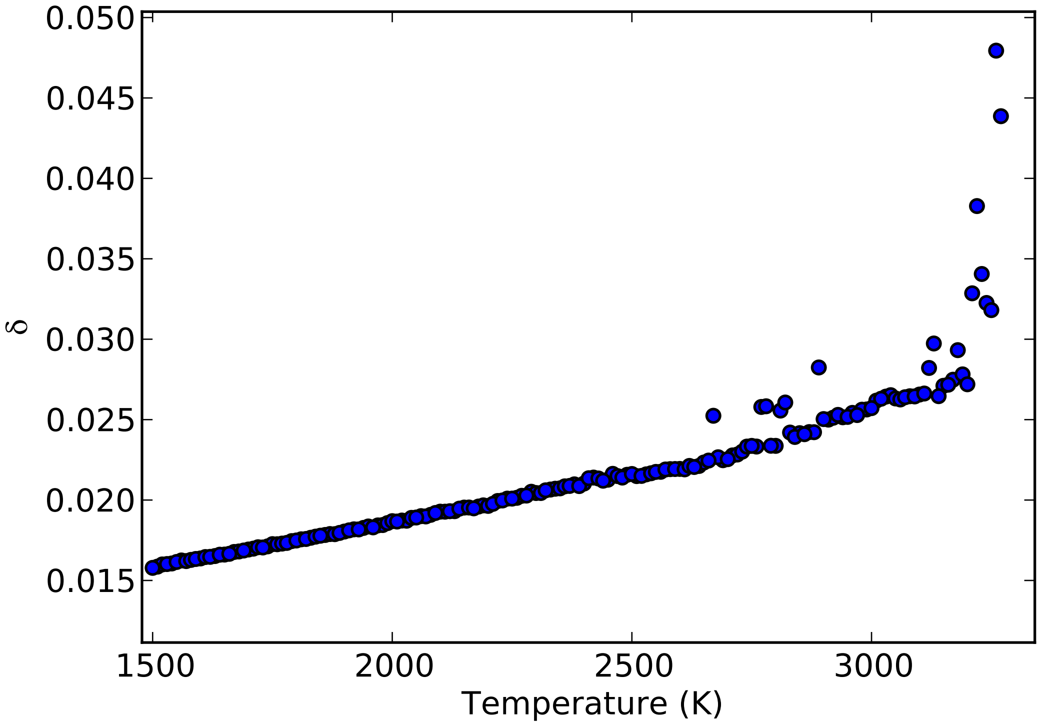

Figure 10 LINDEMANN INDEX FOR 6 EVACUATED ALUMINUM ATOMS

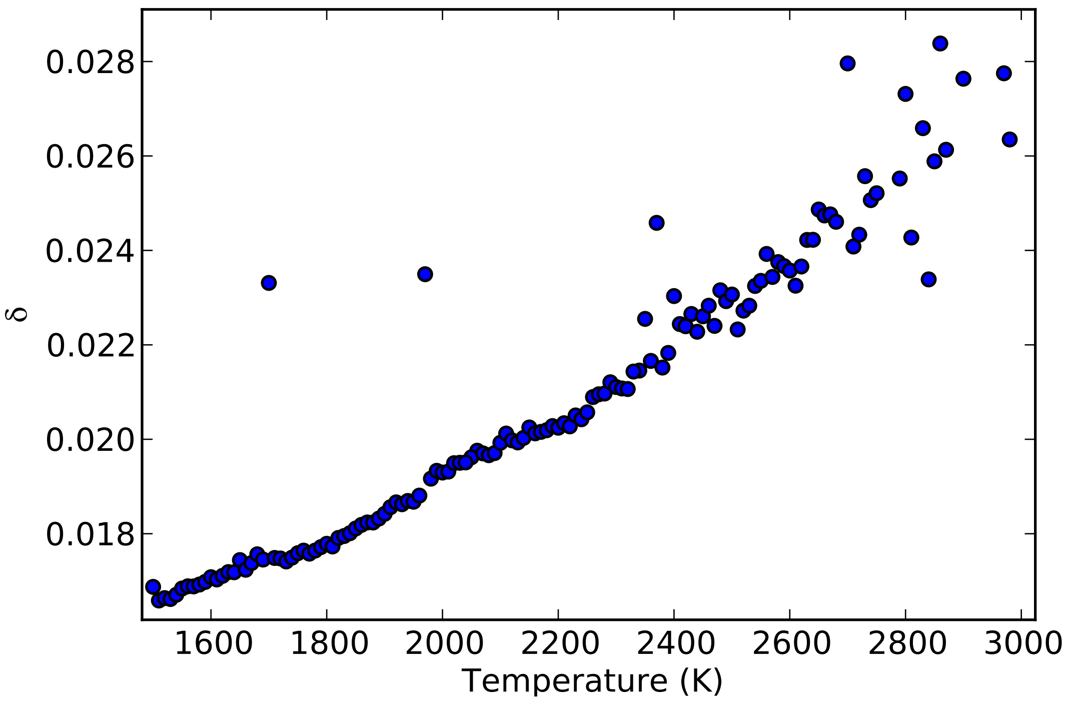

Figure 11 LINDEMANN INDEX FOR 15 EVACUATED ALUMINUM ATOMS

CHAPTER V

THERMAL CONDUCTIVITY FROM NON-EQUILIBRIUM SIMULATIONS

The simulations described in this section measure thermal conductivity in the [100] direction. Since YAG is arranged in a cubic structure, its thermal conductivity is given by a diagonal tensor

$$\mathbf{k}_{ij} = \begin{vmatrix} k & 0 & 0 \\ 0 & k & 0 \\ 0 & 0 & k \end{vmatrix}. \tag{11}$$

Therefore, the thermal conductivity tensor is fully described by this one measure.

An effective thermal conductivity, k, is obtained by imposing a temperature gradient so that Fourier's law of conduction in one dimension may be evaluated directly. In one set of simulations the temperature gradient is specified directly. In another set of simulations, the temperature gradient is specified indirectly by specifiying a temperature difference between each side of a slab. In either case, the thermal conductivity is calculated as

$$k = -\frac{\dot{q}_x}{A_c}\frac{dx}{dT}, \tag{12}$$

where $\dot{q}$ is the amount of energy added to the hot reservoir or removed from the cold reservoir to maintain the temperature. The derivative is the inverse of the temperature gradient that is a result of the simulation.

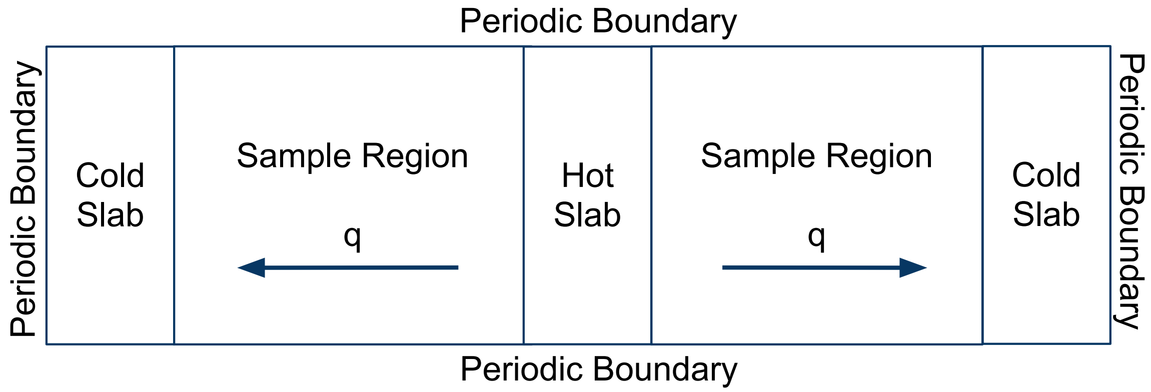

Figure 12 NON-EQUILIBRIUM MD CONFIGURATION

5.1 Simulation Setup

Two temperature gradients were imposed in a symmetric configuration, as represented in Figure 12, since the simulations feature periodic boundary conditions. This configuration ensures that the hot slab is not exposed to the cold slab at the simulation boundary.

For all simulations, the lattice spacing was 12.43 Å. Furthermore, the hot slab was 8×1×1 (L×H×D) unit cells, while each cold slab was 4×1×1 unit cells. As such, the atoms in each 4×1×1 region were assigned to four thermostats. The thermostats were applied to the hot slab and cold slab such that the average temperature of the temperature gradient was fixed at 300 K.

Simulations were varied in the size of the sample region, in random aluminum substitution fraction, x, and in the temperature difference between the hot and cold themostats, ΔT, for one set of simulations. The sample sizes were 4, 6, 8, 12, 16, 24, and 32; x was 0, 50, and 100 percent; and ΔT was 40, 100, and 150 K for a total of 63 simulations. In the other set of simulations, instead of specifiying ΔT, the temperature gradients were 0.4, 0.6, 0.8, and 1 K per lattice constant. The other variables were the same giving 84 simulations.

An initial linear temperature gradient was imposed over the sample region with a random gaussian velocity distribution. The temperature gradient is then maintained by the

Table 4 THERMAL CONDUCTIVITY SIMULATION SETTINGS

timestep	.001 ps
exp. term cutoff	4.0 Å
Coulombic term cutoff	7.0 Å
PPPM precision	10^{-5}
thermostat damping	1 ps

thermostats for 10^6 timsteps (10 ns). The region controlled by the thermostats are time-integrated with the NVT integrator, while the sample region was time-integrated with the NVE integrator.

During the run, the average temperature in each unit cell is calculated every 10 timesteps. The time average is then recorded for 100 of these calculations. Meanwhile, the data from the thermostats are recorded every 1000 timesteps. The data from the thermostats represent the cumulative energy change of the system due to their action. In particular, here it represents an energy addition or subtraction due to the flow of energy from the hot slab to the cold slab.

Table 4 lists the simulation settings. The cutoffs apply to the two-body potential (Equation 1).

5.2 Results and Discussion

A linear temperature profile is exhibited in all simulations when averaged over their run time including systems of 4 unit cells wide as in Figure 13.

To calculate thermal conductivity according to Fourier's law (Equation 12), the slope from a linear fit to the sample region from each side of the temperature profile is averaged to obtain $\Delta x/(A_c \Delta T)$. In addition, $\dot{q}_x = \Delta E/\Delta t$ where ΔE is represented by the energy difference of E_{ts}^{avg} between the start and end of the simulation in steady state and A_c is the sample cross-sectional area.

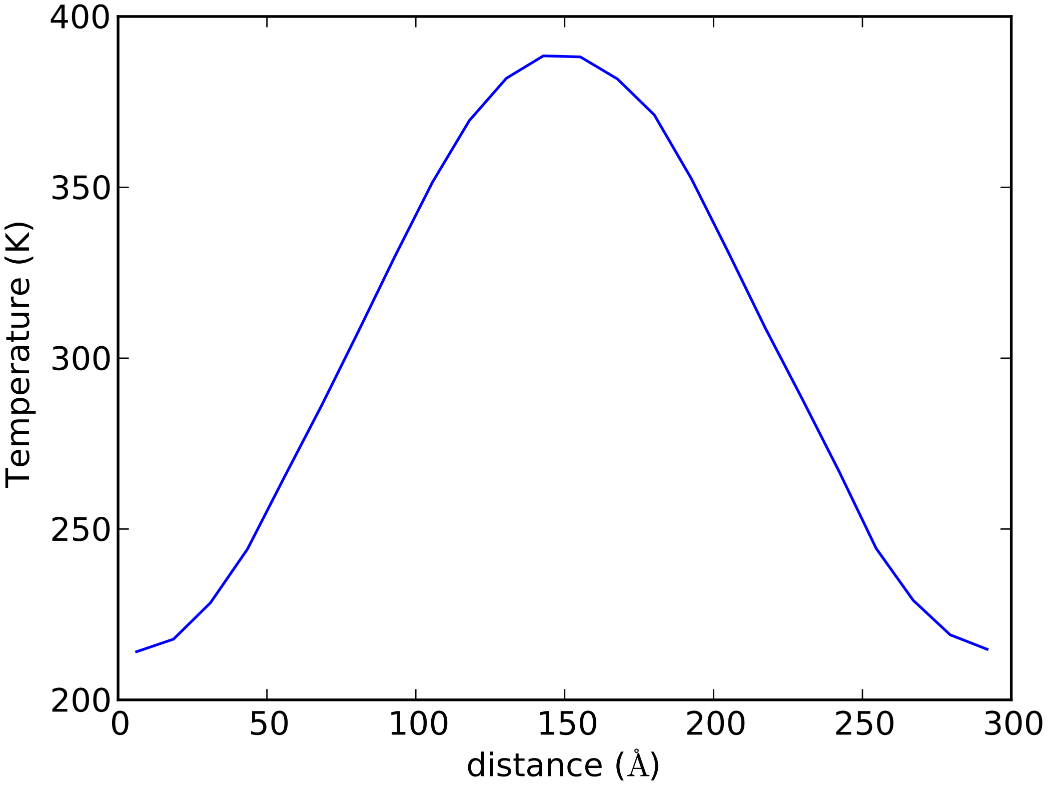

Figure 13 AVERAGE TEMPERATURE PROFILE FOR A SAMPLE SIZE 4 UNIT CELLS WIDE, WITH 50% Al SUBSTITUTION, ΔT=150 K

Figure 14 shows the results of the thermal conductivity calculation for the set of simulations specifying ΔT while Figure 15 shows the calculations for the set of simulations specifiying temperature gradient. The error bars represent the standard deviation in calculating the thermal conductivity for different temperature gradients or temperature differences. For pure YAG, the trend is approaching (the lower end of) its accepted bulk value of 10 W·m^{-1}K^{-1} [12]. All trends of a certain substitution fraction (varying system size) can be explained by simplified kinetic theory: the relationship between thermal conductivity, specific heat capacity, group velocity, and mean free path is given by

$$k = \frac{1}{3} v \Lambda c_v. \tag{13}$$

Because system size is the only variable, the decrease in thermal conductivity with decreasing system size is due to either decreased mean free path or a reduction in available vibrational modes for heat transfer. Long wavelength modes are more efficient at transferring energy, so in systems that involve only 4 unit cells, only 4 acoustic modes (vibrations among unit cells) are possible. Transport, therefore, relies primarily on optical modes (vibrations within unit cells), which are known to be inefficient thermal carriers. Moreover, the acoustic modes will be of short wavelength, whose group velocity will be small. In a system whose size is comparable to the mean free path, reduced thermal conductivity is observed due to the boundary scattering, which reduces the effective mean free path. However, we do not know what the intrinsic mean free path of YAG is; therefore, we can not make strong statements about this effect. Nevertheless, phonon mean free paths are often larger than a few unit cells. So it seems plausible that this effect is present as well.

Meanwhile, group velocity is needed to fully explain the trends for a certain system size (varying substitution fraction). Assuming acoustic phonons are the primary thermal carriers, the lower thermal conductivity of YAG fully substituted with gallium compared to pure YAG can be explained by the difference in their phonon dispersions: as gallium is a heavier element than aluminum, the group velocity of fully substituted YAG will be lower, resulting in lower thermal conductivity. Still, decreased mean free path explains why the thermal conductivity for x=50% is less than x=0% and x=100%. The presence of gallium atoms distributed randomly in the structure represents mass-difference impurity phonon scattering centers that reduce the mean free path of phonons.

The increase in thermal conductivity with increasing system length is expected due to the contribution of longer wavelength phonon modes. However, the trend does not converge over the simulated lengths, and simulating larger lengths is computationally impractical. While an inverse k versus inverse length is a typical approach to estimate the infinite size or bulk thermal conductivity from a reduced system size, others have noted similar difficulties in getting the data to appear linear or converge [22]. These artifacts seems to appear in relatively complex systems such as buckyballs, and YAG certainly qualifies as a complex system. The lack of convergence is probably due to new modes that are introduced as the system size increases rather than simply adding more frequencies of the same modes, which is what happens in smaller, simpler systems. In addition, the mean free path of various modes are likely different. Therefore, as the system size is increased, a host of scattering parameters come into play making the dominant mechanism become non-linear. Nevertheless, the data show a definite trend that appears to converge to expected bulk values for thermal conductivity of YAG.

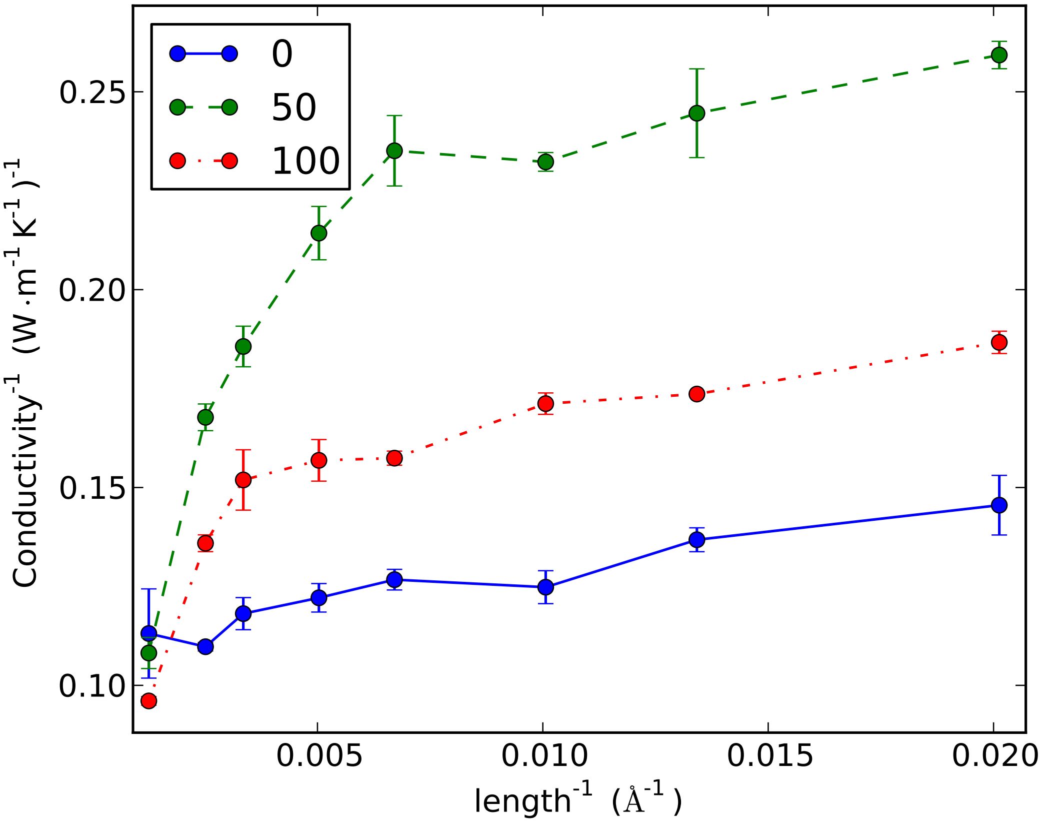

Figure 14 NON-EQUILIBRIUM THERMAL CONDUCTIVITY TRENDS SPECIFYING TEMPERATURE DIFFERENCE FOR x = 0, 50, AND 100%.

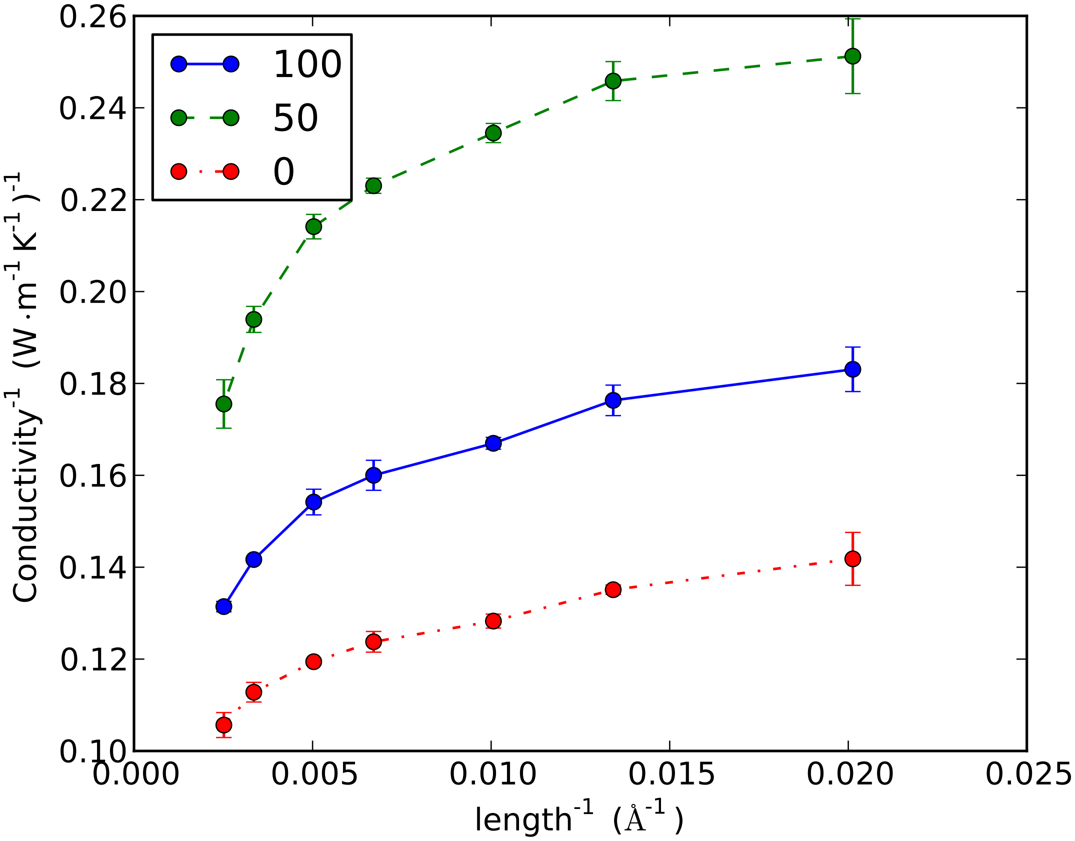

Figure 15 NON-EQUILIBRIUM THERMAL CONDUCTIVITY TRENDS SPECIFYING SPECIFYING TEMPERATURE GRADIENT FOR *x* = 0, 50, AND 100%.

CHAPTER VI

CONCLUSIONS

The results for pure YAG shows good agreement with experimental values for constant-pressure specific heat, thermal expansion, and non-equilibrium thermal conductivity at 300 K, suggesting that the interatomic potentials that were originally developed for mechanical properties are nearly suitable or thermal properties. Of course the quality of the potential must be graded against the quality of the thermal property that needs to be estimated. For order of magnitude and trends, we are confident that these potentials are suitable.

The melting of YAG is not a well-defined process occuring over a wide temperature range. Modeling the surface of YAG, where interatomic potentials are not readily available, is needed to better simulate its melting process. In this case, new potentials are likely needed, so defects can be treated more rigorously. Consideration of this relationship is important in high temperature-transient applications.

The relationship between thermal properties and and aluminum substitution may not be robust. Although qualitative trends were established and corroborated, the simulations do not consider slight modifications to the interatomic potentials due to changes in bonding between gallium and oxygen versus aluminum and oxygen. These changes in bonding could be important, although the foregoing analysis was unable to discern whether the interatomic potentials were sufficient. Therefore, experimental thermal properties for gallium-substituted YAG are needed to confirm the YAG model used in this work.

APPENDIX

LAMMPS SOURCE MODIFICATION

In order to compute three-body energies and their forces for this work, code for two potentials was added based on the included Stillinger-Weber potentials (composed of two and three-body interactions). One implemented the three-body interaction formed by Equation 2 and Equation 3 while zeroing the original two-body interaction. The other involved only zeroing the two-body part of the Stillinger-Weber potentials since the three-body interaction formed by Equation 2 and Equation 4 is compatible with the original code. The following code shows the implementation of Equations 2 and 3 ('mbmh' potential in LAMMPS script).

```
/*------------------------------------------------------------------------
   LAMMPS - Large-scale Atomic/Molecular Massively Parallel Simulator
   http://lammps.sandia.gov, Sandia National Laboratories
   Steve Plimpton, sjplimp@sandia.gov

   Copyright (2003) Sandia Corporation.  Under the terms of Contract
   DE-AC04-94AL85000 with Sandia Corporation, the U.S. Government
       retains
   certain rights in this software.  This software is distributed under
   the GNU General Public License.

   See the README file in the top-level LAMMPS directory.
------------------------------------------------------------------------
   */

/* ----------------------------------------------------------------------
   Contributing author: Aidan Thompson (SNL)
------------------------------------------------------------------------
   */

#include ”math.h”
#include ”stdio.h”
#include ”stdlib.h”
#include ”string.h”
#include ”pair_mbmh.h”
#include ”atom.h”
#include ”neighbor.h”
#include ”neigh_request.h”
#include ”force.h”
#include ”comm.h”
#include ”memory.h”
#include ”neighbor.h”
#include ”neigh_list.h”
#include ”memory.h”
#include ”error.h”
```

```
using namespace LAMMPS_NS;

#define MAXLINE 1024
#define DELTA 4

/* ---------------------------------------------------------------------- */

PairMBMH::PairMBMH(LAMMPS *lmp) : Pair(lmp)
{
  single_enable = 0;
  one_coeff = 1;

  nelements = 0;
  elements = NULL;
  nparams = 0;
  maxparam = 0;
  params = NULL;
  elem2param = NULL;
}

/* ----------------------------------------------------------------------
   check if allocated, since class can be destructed when incomplete
------------------------------------------------------------------------
    */

PairMBMH::~PairMBMH()
{
  if (elements)
    for (int i = 0; i < nelements; i++) delete [] elements[i];
  delete [] elements;
  memory->sfree(params);
  memory->destroy_3d_int_array(elem2param);

  if (allocated) {
    memory->destroy_2d_int_array(setflag);
    memory->destroy_2d_double_array(cutsq);
    delete [] map;
  }
}

/* ---------------------------------------------------------------------- */

void PairMBMH::compute(int eflag, int vflag)
{
  int i,j,k,ii,jj,kk,inum,jnum,jnumm1,itag,jtag;
  int itype,jtype,ktype,ijparam,ikparam,ijkparam;
  double xtmp,ytmp,ztmp,delx,dely,delz,evdwl,fpair;
  double rsq,rsq1,rsq2;
  double delr1[3],delr2[3],fj[3],fk[3];
  int *ilist,*jlist,*numneigh,**firstneigh;

  evdwl = 0.0;
  if (eflag || vflag) ev_setup(eflag,vflag);
  else evflag = vflag_fdotr = 0;
```

```
double **x = atom->x;
double **f = atom->f;
int *tag = atom->tag;
int *type = atom->type;
int nlocal = atom->nlocal;
int newton_pair = force->newton_pair;

inum = list->inum;
ilist = list->ilist;
numneigh = list->numneigh;
firstneigh = list->firstneigh;

// loop over full neighbor list of my atoms

for (ii = 0; ii < inum; ii++) {
  i = ilist[ii];
  itag = tag[i];
  itype = map[type[i]];
  xtmp = x[i][0];
  ytmp = x[i][1];
  ztmp = x[i][2];

  // two-body interactions, skip half of them

  jlist = firstneigh[i];
  jnum = numneigh[i];

  for (jj = 0; jj < jnum; jj++) {
    j = jlist[jj];
    jtag = tag[j];

    if (itag > jtag) {
      if ((itag+jtag) % 2 == 0) continue;
    } else if (itag < jtag) {
      if ((itag+jtag) % 2 == 1) continue;
    } else {
      if (x[j][2] < ztmp) continue;
      else if (x[j][2] == ztmp && x[j][1] < ytmp) continue;
      else if (x[j][2] == ztmp && x[j][1] == ytmp && x[j][0] < xtmp)
          continue;
    }

    jtype = map[type[j]];

    delx = xtmp - x[j][0];
    dely = ytmp - x[j][1];
    delz = ztmp - x[j][2];
    rsq = delx*delx + dely*dely + delz*delz;

    ijparam = elem2param[itype][jtype][jtype];
    if (rsq > params[ijparam].cutsq) continue;

    twobody(&params[ijparam],rsq,fpair,eflag,evdwl);
    f[i][0] += delx*fpair;
    f[i][1] += dely*fpair;
    f[i][2] += delz*fpair;
    f[j][0] -= delx*fpair;
    f[j][1] -= dely*fpair;
    f[j][2] -= delz*fpair;

    if (evflag) ev_tally(i,j,nlocal,newton_pair,
                         evdwl,0.0,fpair,delx,dely,delz);
  }

  jnumm1 = jnum - 1;

  for (jj = 0; jj < jnumm1; jj++) {
    j = jlist[jj];
    jtype = map[type[j]];
    ijparam = elem2param[itype][jtype][jtype];
    delr1[0] = x[j][0] - xtmp;
    delr1[1] = x[j][1] - ytmp;
    delr1[2] = x[j][2] - ztmp;
    rsq1 = delr1[0]*delr1[0] + delr1[1]*delr1[1] + delr1[2]*delr1[2];
    if (rsq1 > params[ijparam].cutsq) continue;

    for (kk = jj+1; kk < jnum; kk++) {
      k = jlist[kk];
      ktype = map[type[k]];
      ikparam = elem2param[itype][ktype][ktype];
      ijkparam = elem2param[itype][jtype][ktype];

      delr2[0] = x[k][0] - xtmp;
      delr2[1] = x[k][1] - ytmp;
      delr2[2] = x[k][2] - ztmp;
      rsq2 = delr2[0]*delr2[0] + delr2[1]*delr2[1] + delr2[2]*delr2
          [2];
      if (rsq2 > params[ikparam].cutsq) continue;

      threebody(&params[ijparam],&params[ikparam],&params[ijkparam],
                rsq1,rsq2,delr1,delr2,fj,fk,eflag,evdwl);

      f[i][0] -= fj[0] + fk[0];
      f[i][1] -= fj[1] + fk[1];
      f[i][2] -= fj[2] + fk[2];
      f[j][0] += fj[0];
      f[j][1] += fj[1];
      f[j][2] += fj[2];
      f[k][0] += fk[0];
      f[k][1] += fk[1];
      f[k][2] += fk[2];

      if (evflag) ev_tally3(i,j,k,evdwl,0.0,fj,fk,delr1,delr2);
    }
  }
}
```

```
  if (vflag_fdotr) virial_compute();
}

/* ---------------------------------------------------------------------- */

void PairMBMH::allocate()
{
  allocated = 1;
  int n = atom->ntypes;

  setflag = memory->create_2d_int_array(n+1,n+1,”pair:setflag”);
  cutsq = memory->create_2d_double_array(n+1,n+1,”pair:cutsq”);

  map = new int[n+1];
}

/* ----------------------------------------------------------------------
   global settings
------------------------------------------------------------------------- */

void PairMBMH::settings(int narg, char **arg)
{
  if (narg != 0) error->all(”Illegal pair_style command”);
}

/* ----------------------------------------------------------------------
   set coeffs for one or more type pairs
------------------------------------------------------------------------- */

void PairMBMH::coeff(int narg, char **arg)
{
  int i,j,n;

  if (!allocated) allocate();

  if (narg != 3 + atom->ntypes)
    error->all(”Incorrect args for pair coefficients”);

  // insure I,J s are * *

  if (strcmp(arg[0],”*”) != 0 || strcmp(arg[1],”*”) != 0)
    error->all(”Incorrect args for pair coefficients”);

  // read args that map atom types to elements in potential file
  // map[i] = which element the Ith atom type is, -1 if NULL
  // nelements = # of unique elements
  // elements = list of element names

  if (elements) {
    for (i = 0; i < nelements; i++) delete [] elements[i];
    delete [] elements;
  }
  elements = new char*[atom->ntypes];
  for (i = 0; i < atom->ntypes; i++) elements[i] = NULL;

  nelements = 0;
  for (i = 3; i < narg; i++) {
    if (strcmp(arg[i],"NULL") == 0) {
      map[i-2] = -1;
      continue;
    }
    for (j = 0; j < nelements; j++)
      if (strcmp(arg[i],elements[j]) == 0) break;
    map[i-2] = j;
    if (j == nelements) {
      n = strlen(arg[i]) + 1;
      elements[j] = new char[n];
      strcpy(elements[j],arg[i]);
      nelements++;
    }
  }

  // read potential file and initialize potential parameters

  read_file(arg[2]);
  setup();

  // clear setflag since coeff() called once with I,J = * *

  n = atom->ntypes;
  for (int i = 1; i <= n; i++)
    for (int j = i; j <= n; j++)
      setflag[i][j] = 0;

  // set setflag i,j for type pairs where both are mapped to elements

  int count = 0;
  for (int i = 1; i <= n; i++)
    for (int j = i; j <= n; j++)
      if (map[i] >= 0 && map[j] >= 0) {
        setflag[i][j] = 1;
        count++;
      }

  if (count == 0) error->all("Incorrect args for pair coefficients");
}

/* ----------------------------------------------------------------------
   init specific to this pair style
------------------------------------------------------------------------- */

void PairMBMH::init_style()
{
  if (atom->tag_enable == 0)
    error->all("Pair style Stillinger-Weber requires atom IDs");
  if (force->newton_pair == 0)
    error->all("Pair style Stillinger-Weber requires newton pair on");
  // need a full neighbor list

  int irequest = neighbor->request(this);
  neighbor->requests[irequest]->half = 0;
  neighbor->requests[irequest]->full = 1;
}

/* ----------------------------------------------------------------------
   init for one type pair i,j and corresponding j,i
------------------------------------------------------------------------
   */

double PairMBMH::init_one(int i, int j)
{
  if (setflag[i][j] == 0) error->all("All pair coeffs are not set");

  return cutmax;
}

/* ---------------------------------------------------------------------- */

void PairMBMH::read_file(char *file)
{
  int params_per_line = 14;
  char **words = new char*[params_per_line+1];

  memory->sfree(params);
  params = NULL;
  nparams = 0;

  // open file on proc 0

  FILE *fp;
  if (comm->me == 0) {
    fp = fopen(file,"r");
    if (fp == NULL) {
      char str[128];
      sprintf(str,"Cannot open Stillinger-Weber potential file %s",file)
          ;
      error->one(str);
    }
  }

  // read each set of params from potential file
  // one set of params can span multiple lines
  // store params if all 3 element tags are in element list

  int n,nwords,ielement,jelement,kelement;
  char line[MAXLINE],*ptr;
  int eof = 0;

  while (1) {
    if (comm->me == 0) {
```

```
  ptr = fgets(line,MAXLINE,fp);
  if (ptr == NULL) {
    eof = 1;
    fclose(fp);
  } else n = strlen(line) + 1;
}
MPI_Bcast(&eof,1,MPI_INT,0,world);
if (eof) break;
MPI_Bcast(&n,1,MPI_INT,0,world);
MPI_Bcast(line,n,MPI_CHAR,0,world);

// strip comment, skip line if blank

if (ptr = strchr(line,’#’)) *ptr = ’\0’;
nwords = atom->count_words(line);
if (nwords == 0) continue;

// concatenate additional lines until have params_per_line words

while (nwords < params_per_line) {
  n = strlen(line);
  if (comm->me == 0) {
    ptr = fgets(&line[n],MAXLINE-n,fp);
    if (ptr == NULL) {
      eof = 1;
      fclose(fp);
    } else n = strlen(line) + 1;
  }
  MPI_Bcast(&eof,1,MPI_INT,0,world);
  if (eof) break;
  MPI_Bcast(&n,1,MPI_INT,0,world);
  MPI_Bcast(line,n,MPI_CHAR,0,world);
  if (ptr = strchr(line,’#’)) *ptr = ’\0’;
  nwords = atom->count_words(line);
}

if (nwords != params_per_line)
  error->all(”Incorrect format in Stillinger-Weber potential file”);

// words = ptrs to all words in line

nwords = 0;
words[nwords++] = strtok(line,” \t\n\r\f”);
while (words[nwords++] = strtok(NULL,” \t\n\r\f”)) continue;

// ielement,jelement,kelement = 1st args
// if all 3 args are in element list, then parse this line
// else skip to next entry in file

for (ielement = 0; ielement < nelements; ielement++)
  if (strcmp(words[0],elements[ielement]) == 0) break;
if (ielement == nelements) continue;
for (jelement = 0; jelement < nelements; jelement++)
  if (strcmp(words[1],elements[jelement]) == 0) break;
```

```
    if (jelement == nelements) continue;
    for (kelement = 0; kelement < nelements; kelement++)
      if (strcmp(words[2],elements[kelement]) == 0) break;
    if (kelement == nelements) continue;

    // load up parameter settings and error check their values

    if (nparams == maxparam) {
      maxparam += DELTA;
      params = (Param *) memory->srealloc(params,maxparam*sizeof(Param),
                                         ”pair:params”);
    }

    params[nparams].ielement = ielement;
    params[nparams].jelement = jelement;
    params[nparams].kelement = kelement;
    params[nparams].epsilon = atof(words[3]);
    params[nparams].sigma = atof(words[4]);
    params[nparams].littlea = atof(words[5]);
    params[nparams].lambda = atof(words[6]);
    params[nparams].gamma = atof(words[7]);
    params[nparams].costheta = atof(words[8]);
    params[nparams].biga = atof(words[9]);
    params[nparams].bigb = atof(words[10]);
    //params[nparams].powerp = atof(words[11]);
    params[nparams].qi = atof(words[11]);
    //params[nparams].powerq = atof(words[12]);
    params[nparams].qj = atof(words[12]);
    params[nparams].tol = atof(words[13]);

    if (params[nparams].epsilon < 0.0 || params[nparams].sigma < 0.0 ||
        params[nparams].littlea < 0.0 || params[nparams].lambda < 0.0 ||
        params[nparams].gamma < 0.0 || params[nparams].biga < 0.0 ||
        //params[nparams].bigb < 0.0 || params[nparams].powerp < 0.0 ||
        params[nparams].bigb < 0.0 ||
        //params[nparams].qi < 0.0 || i might want a neg value
        //params[nparams].powerq < 0.0 || params[nparams].tol < 0.0)
     // params[nparams].qj < 0.0 || i might want a neg value
       params[nparams].tol < 0.0)
      error->all(”Illegal Stillinger-Weber parameter”);

    nparams++;
  }

  delete [] words;
}

/* ---------------------------------------------------------------------- */

void PairMBMH::setup()
{
  int i,j,k,m,n;
  double rtmp;
```

```
// set elem2param for all triplet combinations
// must be a single exact match to lines read from file
// do not allow for ACB in place of ABC

if (elem2param) memory->destroy_3d_int_array(elem2param);
elem2param = memory->create_3d_int_array(nelements,nelements,nelements
    ,
                                          "pair:elem2param");

for (i = 0; i < nelements; i++)
  for (j = 0; j < nelements; j++)
    for (k = 0; k < nelements; k++) {
      n = -1;
      for (m = 0; m < nparams; m++) {
        if (i == params[m].ielement && j == params[m].jelement &&
            k == params[m].kelement) {
          if (n >= 0) error->all("Potential file has duplicate entry")
              ;
          n = m;
        }
      }
      if (n < 0) error->all("Potential file is missing an entry");
      elem2param[i][j][k] = n;
    }


// compute parameter values derived from inputs

// set cutsq using shortcut to reduce neighbor list for accelerated
// calculations. cut must remain unchanged as it is a potential
   parameter
// (cut = a*sigma)

for (m = 0; m < nparams; m++) {
  params[m].cut = params[m].sigma*params[m].littlea;

  rtmp = params[m].cut;
  if (params[m].tol > 0.0) {
    if (params[m].tol > 0.01) params[m].tol = 0.01;
    if (params[m].gamma < 1.0)
      rtmp = rtmp +
        params[m].gamma * params[m].sigma / log(params[m].tol);
    else rtmp = rtmp +
           params[m].sigma / log(params[m].tol);
  }
  params[m].cutsq = rtmp * rtmp;

  params[m].sigma_gamma = params[m].sigma*params[m].gamma;
  params[m].lambda_epsilon = params[m].lambda*params[m].epsilon;
  //params[m].lambda_epsilon2 = 2.0*params[m].lambda*params[m].epsilon
     ;

  params[m].erfc= 14.39965173*params[m].qi*params[m].qj;
  params[m].delerfc=16.248267*params[m].qi*params[m].qj/params[m].bigb
```

```
        ;
    params[m].delexp=3.448275862*params[m].biga;

    /*params[m].c1 = params[m].biga*params[m].epsilon *
      params[m].powerp*params[m].bigb *
      pow(params[m].sigma, params[m].powerp);
    params[m].c2 = params[m].biga*params[m].epsilon*params[m].powerq *
      pow(params[m].sigma, params[m].powerq);
    params[m].c3 = params[m].biga*params[m].epsilon*params[m].bigb *
      pow(params[m].sigma, params[m].powerp+1.0);
    params[m].c4 = params[m].biga*params[m].epsilon *     wont need so
        many c's
      pow(params[m].sigma, params[m].powerq+1.0);
    params[m].c5 = params[m].biga*params[m].epsilon*params[m].bigb *
      pow(params[m].sigma, params[m].powerp);
    params[m].c6 = params[m].biga*params[m].epsilon *
      pow(params[m].sigma, params[m].powerq);*/
  }

  // set cutmax to max of all params

  cutmax = 0.0;
  for (m = 0; m < nparams; m++) {
    rtmp = sqrt(params[m].cutsq);
    if (rtmp > cutmax) cutmax = rtmp;
  }
}

/* ---------------------------------------------------------------------- */

void PairMBMH::twobody(Param *param, double rsq, double &fforce,
                        int eflag, double &eng)
{
  //double r, rinvsq, rp, rq, rainv, rainvsq, expsrainv;
  /*original code
  r = sqrt(rsq);
  rinvsq = 1.0/rsq;
  rp = pow(r,-param->powerp);
  rq = pow(r,-param->powerq);
  rainv = 1.0 / (r - param->cut);
  rainvsq = rainv*rainv*r; //why times r?
  expsrainv = exp(param->sigma * rainv);
  fforce = (param->c1*rp - param->c2*rq +
            (param->c3*rp -param->c4*rq) * rainvsq) * expsrainv * rinvsq
                ;
  if (eflag) eng = (param->c5*rp - param->c6*rq) * expsrainv;
  */
  //V=(14.39965173*q1*q2*erfc(r/B)/r+A*exp(-r/(.29)) where A and B are
      factors; and q is a charge
  //-dV/dr=14.39965173*q1*q2*erfc(r/B)/r^2+28.79930346*q1*q2*exp(-r^2/B
      ^2)/(r*sqrt(Pi)*B)+3.448275862*A*exp(-3.448275862*r)
  /*below is disabled modified code for separation
  double r;
  double paramerfcrbyB, expnegrbypt29;
  r = sqrt(rsq);
  paramerfcrbyB = param->erfc*erfc(r/param->bigb); expnegrbypt29=exp
     (-3.448275862*r);
  */
  fforce =0;//(1.0/r)*(paramerfcrbyB/rsq+param->delerfc*(exp(-rsq/(param
     ->bigb*param->bigb)))/r+param->delexp*expnegrbypt29);//"extra r" b
     /c it's xied by  length vector in compute()
  if (eflag) eng = 0;//(paramerfcrbyB)/r + param->biga*expnegrbypt29;
}

/* ---------------------------------------------------------------------- */

void PairMBMH::threebody(Param *paramij, Param *paramik, Param *paramijk
   ,
                        double rsq1, double rsq2,
                        double *delr1, double *delr2,
                        double *fj, double *fk, int eflag, double &eng)
{
  double r1,rinvsq1,rainv1,gsrainv1,gsrainvsq1;//,expgsrainv1;
  double r2,rinvsq2,rainv2,gsrainv2,gsrainvsq2;//,expgsrainv2;
  double rinv12,cs,delcs,delcssq,expexp,lambdaepsilonexpexp;
  double cssq,snsq,cscb,cspwr4,sncssq, delsqxsncssq;//added
  double gsterm1,gsterm2, cs3term,cs3term1,cs3term2,cs4term,cs4term1,
     cs4term2;

  r1 = sqrt(rsq1);
  rinvsq1 = 1.0/rsq1;
  rainv1 = 1.0/(r1 - paramij->cut);
  gsrainv1 = paramij->sigma_gamma * rainv1;
  gsrainvsq1 = gsrainv1*rainv1;// /r1;
  //expgsrainv1 = exp(gsrainv1);

  r2 = sqrt(rsq2);
  rinvsq2 = 1.0/rsq2;
  rainv2 = 1.0/(r2 - paramik->cut);
  gsrainv2 = paramik->sigma_gamma * rainv2;
  gsrainvsq2 = gsrainv2*rainv2;// /r2;
  //expgsrainv2 = exp(gsrainv2);

  rinv12 = 1.0/(r1*r2);
  cs = (delr1[0]*delr2[0] + delr1[1]*delr2[1] + delr1[2]*delr2[2]) *
     rinv12;
  cssq= cs*cs;
  snsq=1.0-cssq;
  cscb=cssq*cs;
  cspwr4=cscb*cs;
  sncssq= cssq*snsq;
  delcs = cs - paramijk->costheta;
  delcssq = delcs*delcs;
  delsqxsncssq=delcssq*sncssq;

  lambdaepsilonexpexp = paramijk->lambda_epsilon*exp(gsrainv1+gsrainv2);
       //expgsrainv1*expgsrainv2; //majid:why not exp(a+b)?
  gsterm1 = delsqxsncssq*gsrainvsq1; gsterm2=delsqxsncssq*gsrainvsq2;
  cs4term= 2.0*((delcs*cscb+delcssq*cssq)*snsq-delcssq*cspwr4); cs4term1
      =cs4term/r1; cs4term2=cs4term/r2;
  cs3term= 2.0*((-delcs*cssq-delcssq*cs)*snsq+delcssq*cscb)*
      lambdaepsilonexpexp; cs3term1=cs3term/r1; cs3term2=cs3term/r2;

//f(v)function.  (delcs^2*cs(v)^2*sn^2*gs/(r1(v)-as)^2+((2*delcs*cs(v)
    ^3+2*delcs^2*cs(v)^2)*sn^2-2*delcs^2*cs(v)^4)/r1)*LE*exp^2*Vr1hat
    +((-2*delcs*cs(v)^2-2*delcs^2*cs(v))*sn^2+2*delcs^2*cs(v)^3)*LE*exp
    ^2*Vr2hat/r1
  fj[0] = (gsterm1+cs4term1)*lambdaepsilonexpexp*delr1[0]/r1+cs3term1*
      delr2[0]/r2;
  fj[1] = (gsterm1+cs4term1)*lambdaepsilonexpexp*delr1[1]/r1+cs3term1*
      delr2[1]/r2;
  fj[2] = (gsterm1+cs4term1)*lambdaepsilonexpexp*delr1[2]/r1+cs3term1*
      delr2[2]/r2;

  fk[0] = (gsterm2+cs4term2)*lambdaepsilonexpexp*delr2[0]/r2+cs3term2*
      delr1[0]/r1;
  fk[1] = (gsterm2+cs4term2)*lambdaepsilonexpexp*delr2[1]/r2+cs3term2*
      delr1[1]/r1;
  fk[2] = (gsterm2+cs4term2)*lambdaepsilonexpexp*delr2[2]/r2+cs3term2*
      delr1[2]/r1;

  if (eflag) eng = lambdaepsilonexpexp*delsqxsncssq;
}
```

LAMMPS SOURCE MODIFICATION TEST

Simulation Setup Script

```
label inits
dimension 3
units metal
boundary p p p
newton on
atom_style charge


variable a equal (300*9.67114349e-05)+12.3918
#geometry
#primitive vecs  a1 -.5 .5 .5 a2 .5 -.5 .5 a3 .5 .5 -.5 &
#conventional a1 1 0 0 a2 0 1 0 a3 0 0 1 &
#basis vecs below ,Al(16),A2(24),O(96),Y(24) in order, for conventional
#half of that for primitive
#ERROR Input line too long!! changed input.cpp MAXLINES
#using $a as initial
lattice custom $a &
basis    0.0      0.0     0.0          &
basis    0.5      0       0.5     &
basis    0        0.5     0.5     &
basis    0.5      0.5     0       &
basis    0.75     0.25    0.25    &
basis    0.75     0.75    0.75    &
basis    0.25     0.25    0.75    &
basis    0.25     0.75    0.25    &
basis    0.25     0.75    0.75    &
basis    0.25     0.25    0.25    &
basis    0.75     0.75    0.25    &
basis    0.75     0.25    0.75    &
basis    0.5      0.5     0.5     &
basis    0        0.5     0       &
basis    0.5      0       0       &
basis    0        0       0.5     &
basis    0.375    0       0.25    &
basis    0.125    0       0.75    &
basis    0.625    0.5     0.25    &
basis    0.875    0.5     0.75    &
basis    0.25     0.375   0       &
basis    0.75     0.125   0       &
basis    0.25     0.625   0.5     &
basis    0.75     0.875   0.5     &
basis    0        0.25    0.375   &
basis    0        0.75    0.125   &
basis    0.5      0.25    0.625   &
basis    0.5      0.75    0.875   &
basis    0.75     0.625   0       &
basis    0.75     0.375   0.5     &
basis    0.25     0.875   0       &
basis    0.25     0.125   0.5     &
basis    0.125    0.5     0.25    &
basis   0.875   0       0.25    &
basis   0.375   0.5     0.75    &
basis   0.625   0       0.75    &
basis   0       0.25    0.875   &
basis   0.5     0.25    0.125   &
basis   0       0.75    0.625   &
basis   0.5     0.75    0.375   &
basis   0.28023 0.1011  0.19922 &
basis   0.21977 0.8989  0.69922 &
basis   0.71977 0.6011  0.30078 &
basis   0.78023 0.3989  0.80078 &
basis   0.19922 0.28023 0.1011  &
basis   0.69922 0.21977 0.8989  &
basis   0.30078 0.71977 0.6011  &
basis   0.80078 0.78023 0.3989  &
basis   0.1011  0.19922 0.28023 &
basis   0.8989  0.69922 0.21977 &
basis   0.6011  0.30078 0.71977 &
basis   0.3989  0.80078 0.78023 &
basis   0.8511  0.53023 0.05078 &
basis   0.6489  0.46977 0.55078 &
basis   0.3511  0.96977 0.94922 &
basis   0.1489  0.03023 0.44922 &
basis   0.03023 0.44922 0.1489  &
basis   0.96977 0.94922 0.3511  &
basis   0.46977 0.55078 0.6489  &
basis   0.53023 0.05078 0.8511  &
basis   0.94922 0.3511  0.96977 &
basis   0.44922 0.1489  0.03023 &
basis   0.05078 0.8511  0.53023 &
basis   0.55078 0.6489  0.46977 &
basis   0.71977 0.8989  0.80078 &
basis   0.78023 0.1011  0.30078 &
basis   0.28023 0.3989  0.69922 &
basis   0.21977 0.6011  0.19922 &
basis   0.80078 0.71977 0.8989  &
basis   0.30078 0.78023 0.1011  &
basis   0.69922 0.28023 0.3989  &
basis   0.19922 0.21977 0.6011  &
basis   0.8989  0.80078 0.71977 &
basis   0.1011  0.30078 0.78023 &
basis   0.3989  0.69922 0.28023 &
basis   0.6011  0.19922 0.21977 &
basis   0.1489  0.46977 0.94922 &
basis   0.3511  0.53023 0.44922 &
basis   0.6489  0.03023 0.05078 &
basis   0.8511  0.96977 0.55078 &
basis   0.96977 0.55078 0.8511  &
basis   0.03023 0.05078 0.6489  &
basis   0.53023 0.44922 0.3511  &
basis   0.46977 0.94922 0.1489  &
basis   0.05078 0.6489  0.03023 &
basis   0.55078 0.8511  0.96977 &
basis   0.94922 0.1489  0.46977 &
basis   0.44922 0.3511  0.53023 &
basis   0.78023 0.6011  0.69922 &
basis   0.71977 0.3989  0.19922 &
basis   0.21977 0.1011  0.80078 &
basis   0.28023 0.8989  0.30078 &
basis   0.69922 0.78023 0.6011  &
basis   0.19922 0.71977 0.3989  &
basis   0.80078 0.21977 0.1011  &
basis   0.30078 0.28023 0.8989  &
basis   0.6011  0.69922 0.78023 &
basis   0.3989  0.19922 0.71977 &
basis   0.1011  0.80078 0.21977 &
basis   0.8989  0.30078 0.28023 &
basis   0.3511  0.03023 0.55078 &
basis   0.1489  0.96977 0.05078 &
basis   0.8511  0.46977 0.44922 &
basis   0.6489  0.53023 0.94922 &
basis   0.53023 0.94922 0.6489  &
basis   0.46977 0.44922 0.8511  &
basis   0.96977 0.05078 0.1489  &
basis   0.03023 0.55078 0.3511  &
basis   0.44922 0.8511  0.46977 &
basis   0.94922 0.6489  0.53023 &
basis   0.55078 0.3511  0.03023 &
basis   0.05078 0.1489  0.96977 &
basis   0.21977 0.3989  0.30078 &
basis   0.28023 0.6011  0.80078 &
basis   0.78023 0.8989  0.19922 &
basis   0.71977 0.1011  0.69922 &
basis   0.30078 0.21977 0.3989  &
basis   0.80078 0.28023 0.6011  &
basis   0.19922 0.78023 0.8989  &
basis   0.69922 0.71977 0.1011  &
basis   0.3989  0.30078 0.21977 &
basis   0.6011  0.80078 0.28023 &
basis   0.8989  0.19922 0.78023 &
basis   0.1011  0.69922 0.71977 &
basis   0.6489  0.96977 0.44922 &
basis   0.8511  0.03023 0.94922 &
basis   0.1489  0.53023 0.55078 &
basis   0.3511  0.46977 0.05078 &
basis   0.46977 0.05078 0.3511  &
basis   0.53023 0.55078 0.1489  &
basis   0.03023 0.94922 0.8511  &
basis   0.96977 0.44922 0.6489  &
basis   0.55078 0.1489  0.53023 &
basis   0.05078 0.3511  0.46977 &
basis   0.44922 0.6489  0.96977 &
basis   0.94922 0.8511  0.03023 &
basis   0.125   0       0.25    &
basis   0.375   0       0.75    &
basis   0.875   0.5     0.25    &
basis   0.625   0.5     0.75    &
basis   0.25    0.125   0       &
basis   0.75    0.375   0       &
basis   0.25    0.875   0.5     &
basis   0.75    0.625   0.5     &
basis   0       0.25    0.125   &
basis   0       0.75    0.375   &
basis   0.5     0.25    0.875   &
basis   0.5     0.75    0.625   &
basis   0.875   0       0.75    &
basis   0.625   0       0.25    &
basis   0.125   0.5     0.75    &
basis   0.375   0.5     0.25    &
basis   0.75    0.875   0       &
basis   0.25    0.625   0       &
basis   0.75    0.125   0.5     &
basis   0.25    0.375   0.5     &
basis   0       0.75    0.875   &
basis   0       0.25    0.625   &
basis   0.5     0.75    0.125   &
basis   0.5     0.25    0.375   &

region wholething block 0 1 0 1 0 1 units lattice
create_box 4 wholething #no. is ntypes  ***

#after basis kw 1st no. is basis atm, 2nd is type
create_atoms    1       box     basis   1       1 #creates 160 atoms at
    once
#...will need to spec types
group All id <> 1 16
group Al2 id <> 17 40
group Al union All Al2
group O id <> 41 136
group Y id <> 137 160
set group Al type 1
set group O type 2
set group Y type 3



#Atom properties
mass 1 26.98 #Al
mass 2 15.999 #O
mass 3 88.906 #Y
mass 4 69.7 #Dopant Ga for Al

#can add a dopant as mass 4
set group O charge −2.0 #O (−2)
set group Al charge 3.0 #Al (+3) ...the doped−for atom
set group Y charge 3.0 #Y (+3)

include runreqs.lmpin

#looking at one side of sim box
variable dx equal 2
variable ly equal 2
variable lz equal 2
```

```
#create slab to be extruded
replicate 1 2 2
replicate 2 1 1

#random substitution
set group Al type/fraction 4 0 1234  # ***
group Ga type 4
set group Ga charge 3.0 #Ga

variable T equal 300
velocity all create 300 1234 dist gaussian

reset_timestep 0
write_restart r.restart
```

Interatomic Potentials

runreqs.lmpin

```

timestep .002
kspace_style pppm .00001 #usually at .0001 but need some speed, .00001
    conserves but slow

#potentials block
pair_style hybrid/overlay born/coul/long 4.0 7.0 mbmh mbmh2
##2body params
#w/o Ga
pair_coeff 2 2 born/coul/long 2449.44 0.2907 0 0 0
pair_coeff 1 2 born/coul/long 1740.31 0.2907 0 0 0
pair_coeff 2 3 born/coul/long 1250.85 0.3497 0 0 0
pair_coeff 1 1 born/coul/long 312.11 0.071124 0 0 0
pair_coeff 3 3 born/coul/long 245.14 0.071124 0 0 0
pair_coeff 1 3 born/coul/long 256.55 0.071124 0 0 0
#w Ga, replaced the 1s w 4s  (but smaller no. has to be 1st
pair_coeff 2 4 born/coul/long 1740.31 0.2907 0 0 0
pair_coeff 4 4 born/coul/long 312.11 0.071124 0 0 0
pair_coeff 1 4 born/coul/long 312.11 0.071124 0 0 0
pair_coeff 3 4 born/coul/long 256.55 0.071124 0 0 0
##3body params
pair_coeff * * mbmh yag.mbmh A O Y A #maps 4th type to A
pair_coeff * * mbmh2 yag.mbmh2 A O Y A
#recovery from a binary w/ different no of procs
neigh_modify delay 0 every 1 check yes
```

3-Body parameters (yag.mbmh)

```
#3body only for certain kinds and only for XYY
#(a*(sigma=1)) is rij0, lambda*(epsilon=1) is just lambda
# KEEP IT 1 FOR ALL EVEN IF NO THREE BODY INTERACTION!
#b/c same param used in 2body and we dont want to zero that

A A A 1.0   1.0   5.0   0.0   0.0   -0.333333333333 #cut@5
          312.11   14.06   3.0   3.0  0.0
```

```
A A O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A A Y 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A O A 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A O O 1.0   1.0   3.0   149.324   2.8   −0.333333333333# O–Al–O
          1740.31   3.44   −2.0   3.0  0.0
A O Y 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A Y A 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A Y O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A Y Y 1.0   1.0   5.0   0.0   0.0   −0.333333333333 #no 3b, cut@5
          256.55   14.06   3.0   3.0  0.0
O A A 1.0   1.0   3.0   0.0   0.0   −0.333333333333 #3b in 2nd file
          1740.31   3.44   −2.0   3.0  0.0
O A O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O A Y 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #3b in 2nd file , no 2b
          0.0   0.0   0.0   0.0  0.0
O O A 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O O O 1.0   1.0   3.0   0.0   0.0   −0.333333333333 #cut@3
          2449.44   3.44   −2.0   −2.0  0.0
O O Y 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O Y A 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #3b in 2nd file , no 2b
          0.0   0.0   0.0   0.0  0.0
O Y O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O Y Y 1.0   1.0   3.0   0.0   0.0   −0.333333333333 #3b in 2nd file
          1250.85   2.86   −2.0   3.0  0.0
Y A A 1.0   1.0   5.0   0.0   0.0   −0.333333333333 #no 3b, cut@5
          256.55   14.06   3.0   3.0  0.0
Y A O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y A Y 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y O A 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y O O 1.0   1.0   3.0   168.25   2.8   −0.333333333333 #
          1250.85   2.86   3.0   −2.0  0.0
Y O Y 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y Y A 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y Y O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y Y Y 1.0   1.0   3.0   0.0   0.0   −0.333333333333 #no 3b, cut@3
          245.14   14.06   3.0   3.0  0.0
```

3-Body parameters (yag.mbmh2)

```
#3body only for certain kinds and only for XYY
#(a*(sigma=1)) is rij0, lambda*(epsilon=1) is just lambda
# KEEP IT 1 FOR ALL EVEN IF NO THREE BODY INTERACTION!
#actually zero the two-body interaction here

A A A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
A A O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
A A Y 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
A O A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
A O O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
A O Y 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
A Y A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
A Y O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
A Y Y 1.0  1.0  5.0  0.0  0.0  -0.333333333333 #no 3b, cut@5
          256.55  14.06  3.0  3.0 0.0
O A A 1.0  1.0  2.6  6.242  2.0  -0.333333333333 #3b in this file
          0.0  0.0  0.0  0.0 0.0
O A O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
O A Y 1.0  1.0  2.6  6.242  2.0  -0.333333333333 #3b in this file (symm
    ?), no 2b
          0.0  0.0  0.0  0.0 0.0
O O A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
O O O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
O O Y 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
O Y A 1.0  1.0  2.6  6.242  2.0  -0.333333333333 #3b in this file (symm
    ?), no 2b
          0.0  0.0  0.0  0.0 0.0
O Y O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
O Y Y 1.0  1.0  2.6  6.242  2.0  -0.333333333333 #3b in 2nd file, no 2b
          0.0  0.0  0.0  0.0 0.0
Y A A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
Y A O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
Y A Y 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
Y O A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
Y O O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
          0.0  0.0  0.0  0.0 0.0
```

```
Y O Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0 0.0
Y Y A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0 0.0
Y Y O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0 0.0
Y Y Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0 0.0
```

Simulation Execution

```
label start
##input
read_restart r.restart
#stuff not in restart file
include runreqs.lmpin

set group Al type 1
set group O type 2
set group Y type 3
set group Ga type 4
group AlGa union Al Ga

variable a equal (300*9.67114349e-05)+12.3918
lattice sc 1 spacing $a $a $a

variable U equal etotal
variable ts equal step
variable dx equal 2
variable ly equal 2
variable lz equal 2
variable Tk equal 300
compute kea all ke/atom

fix nvef all nve

#two stages: 1. thermalize 2. ss while recording
#what stage is the sim at?
if ${ts}<1000000 then ”jump actions.lmpin exe” &
else ”jump actions.lmpin end”

label exe
#just so i can have file names w/ timesteps
fix eof all ave/time 1 1 10 v_U file ${ts}.cumdE.avetime
run 1000000 upto every 1000 ”write_restart r.restart”

label end
```

LAMMPS SCRIPTS FOR LATTICE CONSTANT, SPECIFIC HEAT, THERMAL EXPANSION

Simulation Setup

This script sets up the simulation.

```
label inits
dimension 3
units metal
boundary p p p
newton on
atom_style charge

#geometry
#primitive vecs  a1 -.5 .5 .5 a2 .5 -.5 .5 a3 .5 .5 -.5 &
#conventional a1 1 0 0 a2 0 1 0 a3 0 0 1 &
#basis vecs below ,A1(16),A2(24),O(96),Y(24) in order, for conventional
#half of that for primitive
#ERROR Input line too long!! changed input.cpp MAXLINES
#using 12.0 as initial
lattice custom 12.0 &
basis    0.0      0.0      0.0            &
basis    0.5      0        0.5      &
basis    0        0.5      0.5      &
basis    0.5      0.5      0        &
basis    0.75     0.25     0.25     &
basis    0.75     0.75     0.75     &
basis    0.25     0.25     0.75     &
basis    0.25     0.75     0.25     &
basis    0.25     0.75     0.75     &
basis    0.25     0.25     0.25     &
basis    0.75     0.75     0.25     &
basis    0.75     0.25     0.75     &
basis    0.5      0.5      0.5      &
basis    0        0.5      0        &
basis    0.5      0        0        &
basis    0        0        0.5      &
basis    0.375    0        0.25     &
basis    0.125    0        0.75     &
basis    0.625    0.5      0.25     &
basis    0.875    0.5      0.75     &
basis    0.25     0.375    0        &
basis    0.75     0.125    0        &
basis    0.25     0.625    0.5      &
basis    0.75     0.875    0.5      &
basis    0        0.25     0.375    &
basis    0        0.75     0.125    &
basis    0.5      0.25     0.625    &
basis    0.5      0.75     0.875    &
basis    0.75     0.625    0        &
basis    0.75     0.375    0.5      &
basis    0.25     0.875    0        &
basis    0.25     0.125    0.5      &
basis    0.125    0.5      0.25     &
basis    0.875    0        0.25     &
basis   0.375   0.5     0.75    &
basis   0.625   0       0.75    &
basis   0       0.25    0.875   &
basis   0.5     0.25    0.125   &
basis   0       0.75    0.625   &
basis   0.5     0.75    0.375   &
basis   0.28023 0.1011  0.19922 &
basis   0.21977 0.8989  0.69922 &
basis   0.71977 0.6011  0.30078 &
basis   0.78023 0.3989  0.80078 &
basis   0.19922 0.28023 0.1011  &
basis   0.69922 0.21977 0.8989  &
basis   0.30078 0.71977 0.6011  &
basis   0.80078 0.78023 0.3989  &
basis   0.1011  0.19922 0.28023 &
basis   0.8989  0.69922 0.21977 &
basis   0.6011  0.30078 0.71977 &
basis   0.3989  0.80078 0.78023 &
basis   0.8511  0.53023 0.05078 &
basis   0.6489  0.46977 0.55078 &
basis   0.3511  0.96977 0.94922 &
basis   0.1489  0.03023 0.44922 &
basis   0.03023 0.44922 0.1489  &
basis   0.96977 0.94922 0.3511  &
basis   0.46977 0.55078 0.6489  &
basis   0.53023 0.05078 0.8511  &
basis   0.94922 0.3511  0.96977 &
basis   0.44922 0.1489  0.03023 &
basis   0.05078 0.8511  0.53023 &
basis   0.55078 0.6489  0.46977 &
basis   0.71977 0.8989  0.80078 &
basis   0.78023 0.1011  0.30078 &
basis   0.28023 0.3989  0.69922 &
basis   0.21977 0.6011  0.19922 &
basis   0.80078 0.71977 0.8989  &
basis   0.30078 0.78023 0.1011  &
basis   0.69922 0.28023 0.3989  &
basis   0.19922 0.21977 0.6011  &
basis   0.8989  0.80078 0.71977 &
basis   0.1011  0.30078 0.78023 &
basis   0.3989  0.69922 0.28023 &
basis   0.6011  0.19922 0.21977 &
basis   0.1489  0.46977 0.94922 &
basis   0.3511  0.53023 0.44922 &
basis   0.6489  0.03023 0.05078 &
basis   0.8511  0.96977 0.55078 &
basis   0.96977 0.55078 0.8511  &
basis   0.03023 0.05078 0.6489  &
basis   0.53023 0.44922 0.3511  &
basis   0.46977 0.94922 0.1489  &
basis   0.05078 0.6489  0.03023 &
basis   0.55078 0.8511  0.96977 &
basis   0.94922 0.1489  0.46977 &
basis   0.44922 0.3511  0.53023 &
basis   0.78023 0.6011  0.69922 &
basis   0.71977 0.3989  0.19922 &
basis   0.21977 0.1011  0.80078 &
basis   0.28023 0.8989  0.30078 &
basis   0.69922 0.78023 0.6011  &
basis   0.19922 0.71977 0.3989  &
basis   0.80078 0.21977 0.1011  &
basis   0.30078 0.28023 0.8989  &
basis   0.6011  0.69922 0.78023 &
basis   0.3989  0.19922 0.71977 &
basis   0.1011  0.80078 0.21977 &
basis   0.8989  0.30078 0.28023 &
basis   0.3511  0.03023 0.55078 &
basis   0.1489  0.96977 0.05078 &
basis   0.8511  0.46977 0.44922 &
basis   0.6489  0.53023 0.94922 &
basis   0.53023 0.94922 0.6489  &
basis   0.46977 0.44922 0.8511  &
basis   0.96977 0.05078 0.1489  &
basis   0.03023 0.55078 0.3511  &
basis   0.44922 0.8511  0.46977 &
basis   0.94922 0.6489  0.53023 &
basis   0.55078 0.3511  0.03023 &
basis   0.05078 0.1489  0.96977 &
basis   0.21977 0.3989  0.30078 &
basis   0.28023 0.6011  0.80078 &
basis   0.78023 0.8989  0.19922 &
basis   0.71977 0.1011  0.69922 &
basis   0.30078 0.21977 0.3989  &
basis   0.80078 0.28023 0.6011  &
basis   0.19922 0.78023 0.8989  &
basis   0.69922 0.71977 0.1011  &
basis   0.3989  0.30078 0.21977 &
basis   0.6011  0.80078 0.28023 &
basis   0.8989  0.19922 0.78023 &
basis   0.1011  0.69922 0.71977 &
basis   0.6489  0.96977 0.44922 &
basis   0.8511  0.03023 0.94922 &
basis   0.1489  0.53023 0.55078 &
basis   0.3511  0.46977 0.05078 &
basis   0.46977 0.05078 0.3511  &
basis   0.53023 0.55078 0.1489  &
basis   0.03023 0.94922 0.8511  &
basis   0.96977 0.44922 0.6489  &
basis   0.55078 0.1489  0.53023 &
basis   0.05078 0.3511  0.46977 &
basis   0.44922 0.6489  0.96977 &
basis   0.94922 0.8511  0.03023 &
basis   0.125   0       0.25    &
basis   0.375   0       0.75    &
basis   0.875   0.5     0.25    &
basis   0.625   0.5     0.75    &
basis   0.25    0.125   0       &
basis   0.75    0.375   0       &
basis   0.25    0.875   0.5     &
basis   0.75    0.625   0.5     &
basis   0       0.25    0.125   &
basis   0       0.75    0.375   &
basis   0.5     0.25    0.875   &
basis   0.5     0.75    0.625   &
basis   0.875   0       0.75    &
basis   0.625   0       0.25    &
basis   0.125   0.5     0.75    &
basis   0.375   0.5     0.25    &
basis   0.75    0.875   0       &
basis   0.25    0.625   0       &
basis   0.75    0.125   0.5     &
basis   0.25    0.375   0.5     &
basis   0       0.75    0.875   &
basis   0       0.25    0.625   &
basis   0.5     0.75    0.125   &
basis   0.5     0.25    0.375   &

region wholething block 0 1 0 1 0 1 units lattice
create_box 4 wholething #no. is ntypes  ***

#after basis kw 1st no. is basis atm, 2nd is type
create_atoms    1       box     basis   1       1 #creates 160 atoms at
   once
#... will need to spec types
group Al1 id <> 1 16
group Al2 id <> 17 40
group Al union Al1 Al2
group O id <> 41 136
group Y id <> 137 160
set group Al type 1
set group O type 2
set group Y type 3


#Atom properties
mass 1 26.98 #Al
mass 2 15.999 #O
mass 3 88.906 #Y
mass 4 69.7 #Dopant Ga for Al

#can add a dopant as mass 4
set group O charge −2.0 #O (−2)
set group Al charge 3.0 #Al (+3) ...the doped for atom
set group Y charge 3.0 #Y (+3)


include runreqs.lmpin

fix relaxboxfix all box/relax iso $p
minimize 0 0 1000 1000

replicate $ud $ud $ud
```

```
#random doping
set group Al type/fraction 4 $dp $dseed  # ***
group Ga type 4
set group Ga charge 3.0 #Ga

variable T0 equal 10
velocity all create ${T0} $vseed dist gaussian

reset_timestep 0
write_restart r.restart
```

Interatomic Potentials

runreqs.lmpin

```
timestep .002
kspace_style pppm .0001

pair_style hybrid/overlay born/coul/long 4.0 7.0 mbmh mbmh2
##2body params
#w/o Ga
pair_coeff 2 2 born/coul/long 2449.44 0.2907 0 0 0
pair_coeff 1 2 born/coul/long 1740.31 0.2907 0 0 0
pair_coeff 2 3 born/coul/long 1250.85 0.3497 0 0 0
pair_coeff 1 1 born/coul/long 312.11 0.071124 0 0 0
pair_coeff 3 3 born/coul/long 245.14 0.071124 0 0 0
pair_coeff 1 3 born/coul/long 256.55 0.071124 0 0 0
#w Ga, replaced the 1s w 4s  (but smaller no. has to be 1st
pair_coeff 2 4 born/coul/long 1740.31 0.2907 0 0 0
pair_coeff 4 4 born/coul/long 312.11 0.071124 0 0 0
pair_coeff 1 4 born/coul/long 312.11 0.071124 0 0 0
pair_coeff 3 4 born/coul/long 256.55 0.071124 0 0 0
##3body params
pair_coeff * * mbmh yag.mbmh A O Y A #maps 4th type to A
pair_coeff * * mbmh2 yag.mbmh2 A O Y A

#recovery from a binary w/ diffent no of procs
neigh_modify delay 0 every 1 check yes
```

3-Body parameters (yag.mbmh)

```
#3body only for certain kinds and only for XYY
#(a*(sigma=1)) is rij0, lambda*(epsilon=1) is just lambda
# KEEP IT 1 FOR ALL EVEN IF NO THREE BODY INTERACTION!
#b/c same param used in 2body and we dont want to zero that

A A A 1.0   1.0   5.0   0.0   0.0   -0.333333333333 #cut@5
          312.11   14.06   3.0   3.0  0.0
A A O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A A Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A O A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
```

```
         0.0   0.0   0.0   0.0  0.0
A O O 1.0   1.0   3.0   149.324   2.8   -0.333333333333# O-Al-O
         1740.31   3.44   -2.0   3.0  0.0
A O Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
         0.0   0.0   0.0   0.0  0.0
A Y A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
         0.0   0.0   0.0   0.0  0.0
A Y O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
         0.0   0.0   0.0   0.0  0.0
A Y Y 1.0   1.0   5.0   0.0   0.0   -0.333333333333 #no 3b, cut@5
         256.55   14.06   3.0   3.0  0.0
O A A 1.0   1.0   3.0   0.0   0.0   -0.333333333333 #3b in 2nd file
         1740.31   3.44   -2.0   3.0  0.0
O A O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
         0.0   0.0   0.0   0.0  0.0
O A Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #3b in 2nd file, no 2b
         0.0   0.0   0.0   0.0  0.0
O O A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
         0.0   0.0   0.0   0.0  0.0
O O O 1.0   1.0   3.0   0.0   0.0   -0.333333333333 #cut@3
         2449.44   3.44   -2.0   -2.0  0.0
O O Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
         0.0   0.0   0.0   0.0  0.0
O Y A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #3b in 2nd file, no 2b
         0.0   0.0   0.0   0.0  0.0
O Y O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
         0.0   0.0   0.0   0.0  0.0
O Y Y 1.0   1.0   3.0   0.0   0.0   -0.333333333333 #3b in 2nd file
         1250.85   2.86   -2.0   3.0  0.0
Y A A 1.0   1.0   5.0   0.0   0.0   -0.333333333333 #no 3b, cut@5
         256.55   14.06   3.0   3.0  0.0
Y A O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
         0.0   0.0   0.0   0.0  0.0
Y A Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
         0.0   0.0   0.0   0.0  0.0
Y O A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
         0.0   0.0   0.0   0.0  0.0
Y O O 1.0   1.0   3.0   168.25   2.8   -0.333333333333 #
         1250.85   2.86   3.0   -2.0  0.0
Y O Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
         0.0   0.0   0.0   0.0  0.0
Y Y A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
         0.0   0.0   0.0   0.0  0.0
Y Y O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
         0.0   0.0   0.0   0.0  0.0
Y Y Y 1.0   1.0   3.0   0.0   0.0   -0.333333333333 #no 3b, cut@3
         245.14   14.06   3.0   3.0  0.0
```

3-Body parameters (yag.mbmh2)

```
#3body only for certain kinds and only for XYY
#(a*(sigma=1)) is rij0, lambda*(epsilon=1) is just lambda
# KEEP IT 1 FOR ALL EVEN IF NO THREE BODY INTERACTION!
#actually zero the two-body interaction here
```

```
A A A 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A A O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A A Y 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A O A 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A O O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A O Y 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A Y A 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A Y O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A Y Y 1.0   1.0   5.0   0.0   0.0   −0.333333333333 #no 3b, cut@5
          256.55   14.06   3.0   3.0  0.0
O A A 1.0   1.0   2.6   6.242   2.0   −0.333333333333 #3b in this file
          0.0   0.0   0.0   0.0  0.0
O A O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O A Y 1.0   1.0   2.6   6.242   2.0   −0.333333333333 #3b in this file (symm
    ?), no 2b
          0.0   0.0   0.0   0.0  0.0
O O A 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O O O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O O Y 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O Y A 1.0   1.0   2.6   6.242   2.0   −0.333333333333 #3b in this file (symm
    ?), no 2b
          0.0   0.0   0.0   0.0  0.0
O Y O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O Y Y 1.0   1.0   2.6   6.242   2.0   −0.333333333333 #3b in 2nd file, no 2b
          0.0   0.0   0.0   0.0  0.0
Y A A 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y A O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y A Y 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y O A 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y O O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y O Y 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y Y A 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y Y O 1.0   1.0   0.0   0.0   0.0   −0.333333333333 #no 3b, no 2b
```

```
          0.0   0.0   0.0   0.0  0.0
Y Y Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
```

Simulation Execution

```
##input
read_restart r.restart
#stuff not in restart file
include runreqs.lmpin


#1. input min struct 2. set T0 3. npt for a long time
#4. increase by 100 K or so until TF

variable udXts equal 10000*($ud)^3 #cook time
variable U equal etotal
variable T equal temp
variable P equal press
variable a equal (vol^(1.0/3.0))/$ud
variable T0 equal 10
variable ts equal step

variable initloopno equal floor((${ts}/${udXts}))+1
variable todo equal 30-floor((${ts}/${udXts})) #currloop is floor+1
if ”${todo}==0” then ”jump actions.lmpin bypass”

variable loopvar loop ${todo} #if set to 0 then will get an error
#that i don't care about

#want sim to go to right loop initially from read ts
#assign Ttgt, then continue to loop
label loop
variable currloopno equal ${loopvar}+${initloopno}-1 #a workaround for
    loopvar b/c starts w/ 1

variable Ttgt equal ${T0}+(${currloopno}-1)*100

velocity all scale ${Ttgt}
#use rescale for a step
#on 2nd thought i don't want abrupt v deltas

fix nptfix all npt temp ${Ttgt} ${Ttgt} 1 iso $p $p 10 drag 1

fix avgfix all ave/time 10 100 1000 v_U v_T v_P v_a file ${Ttgt}.${ts}.
    thermoavgs.avetime

variable runfor equal ${udXts}*(${currloopno}) #has to be bigger than
    1000
run ${runfor} upto every 1000 ”write_restart r.restart”
#print out last avged
```

```
next loopvar
jump actions.lmpin loop

label bypass
```

LAMMPS SCRIPTS FOR MELTING TEMPERATURE

Simulation Setup

```

label inits
dimension 3
units metal
boundary p p p
newton on
atom_style charge

#geometry
#primitive vecs  a1 -.5 .5 .5 a2 .5 -.5 .5 a3 .5 .5 -.5 &
#conventional a1 1 0 0 a2 0 1 0 a3 0 0 1 &
#basis vecs below ,A1(16),A2(24),O(96),Y(24) in order, for conventional
#half of that for primitive
#ERROR Input line too long!! changed input.cpp MAXLINES
#using 12.0 as initial
lattice custom 12.0 &
basis   0.0     0.0     0.0         &
basis   0.5     0       0.5     &
basis   0       0.5     0.5     &
basis   0.5     0.5     0       &
basis   0.75    0.25    0.25    &
basis   0.75    0.75    0.75    &
basis   0.25    0.25    0.75    &
basis   0.25    0.75    0.25    &
basis   0.25    0.75    0.75    &
basis   0.25    0.25    0.25    &
basis   0.75    0.75    0.25    &
basis   0.75    0.25    0.75    &
basis   0.5     0.5     0.5     &
basis   0       0.5     0       &
basis   0.5     0       0       &
basis   0       0       0.5     &
basis   0.375   0       0.25    &
basis   0.125   0       0.75    &
basis   0.625   0.5     0.25    &
basis   0.875   0.5     0.75    &
basis   0.25    0.375   0       &
basis   0.75    0.125   0       &
basis   0.25    0.625   0.5     &
basis   0.75    0.875   0.5     &
basis   0       0.25    0.375   &
basis   0       0.75    0.125   &
basis   0.5     0.25    0.625   &
basis   0.5     0.75    0.875   &
basis    0.75     0.625    0        &
basis    0.75     0.375    0.5      &
basis    0.25     0.875    0        &
basis    0.25     0.125    0.5      &
basis    0.125    0.5      0.25     &
basis    0.875    0        0.25     &
basis    0.375    0.5      0.75     &
basis    0.625    0        0.75     &
basis    0        0.25     0.875    &
basis    0.5      0.25     0.125    &
basis    0        0.75     0.625    &
basis    0.5      0.75     0.375    &
basis    0.28023  0.1011   0.19922  &
basis    0.21977  0.8989   0.69922  &
basis    0.71977  0.6011   0.30078  &
basis    0.78023  0.3989   0.80078  &
basis    0.19922  0.28023  0.1011   &
basis    0.69922  0.21977  0.8989   &
basis    0.30078  0.71977  0.6011   &
basis    0.80078  0.78023  0.3989   &
basis    0.1011   0.19922  0.28023  &
basis    0.8989   0.69922  0.21977  &
basis    0.6011   0.30078  0.71977  &
basis    0.3989   0.80078  0.78023  &
basis    0.8511   0.53023  0.05078  &
basis    0.6489   0.46977  0.55078  &
basis    0.3511   0.96977  0.94922  &
basis    0.1489   0.03023  0.44922  &
basis    0.03023  0.44922  0.1489   &
basis    0.96977  0.94922  0.3511   &
basis    0.46977  0.55078  0.6489   &
basis    0.53023  0.05078  0.8511   &
basis    0.94922  0.3511   0.96977  &
basis    0.44922  0.1489   0.03023  &
basis    0.05078  0.8511   0.53023  &
basis    0.55078  0.6489   0.46977  &
basis    0.71977  0.8989   0.80078  &
basis    0.78023  0.1011   0.30078  &
basis    0.28023  0.3989   0.69922  &
basis    0.21977  0.6011   0.19922  &
basis    0.80078  0.71977  0.8989   &
basis    0.30078  0.78023  0.1011   &
basis    0.69922  0.28023  0.3989   &
basis    0.19922  0.21977  0.6011   &
basis    0.8989   0.80078  0.71977  &
basis    0.1011   0.30078  0.78023  &
basis    0.3989   0.69922  0.28023  &
basis    0.6011   0.19922  0.21977  &
basis    0.1489   0.46977  0.94922  &
basis    0.3511   0.53023  0.44922  &
basis    0.6489   0.03023  0.05078  &
basis    0.8511   0.96977  0.55078  &
basis    0.96977  0.55078  0.8511   &
basis    0.03023  0.05078  0.6489   &
basis    0.53023 0.44922 0.3511  &
basis    0.46977 0.94922 0.1489  &
basis    0.05078 0.6489  0.03023 &
basis    0.55078 0.8511  0.96977 &
basis    0.94922 0.1489  0.46977 &
basis    0.44922 0.3511  0.53023 &
basis    0.78023 0.6011  0.69922 &
basis    0.71977 0.3989  0.19922 &
basis    0.21977 0.1011  0.80078 &
basis    0.28023 0.8989  0.30078 &
basis    0.69922 0.78023 0.6011  &
basis    0.19922 0.71977 0.3989  &
basis    0.80078 0.21977 0.1011  &
basis    0.30078 0.28023 0.8989  &
basis    0.6011  0.69922 0.78023 &
basis    0.3989  0.19922 0.71977 &
basis    0.1011  0.80078 0.21977 &
basis    0.8989  0.30078 0.28023 &
basis    0.3511  0.03023 0.55078 &
basis    0.1489  0.96977 0.05078 &
basis    0.8511  0.46977 0.44922 &
basis    0.6489  0.53023 0.94922 &
basis    0.53023 0.94922 0.6489  &
basis    0.46977 0.44922 0.8511  &
basis    0.96977 0.05078 0.1489  &
basis    0.03023 0.55078 0.3511  &
basis    0.44922 0.8511  0.46977 &
basis    0.94922 0.6489  0.53023 &
basis    0.55078 0.3511  0.03023 &
basis    0.05078 0.1489  0.96977 &
basis    0.21977 0.3989  0.30078 &
basis    0.28023 0.6011  0.80078 &
basis    0.78023 0.8989  0.19922 &
basis    0.71977 0.1011  0.69922 &
basis    0.30078 0.21977 0.3989  &
basis    0.80078 0.28023 0.6011  &
basis    0.19922 0.78023 0.8989  &
basis    0.69922 0.71977 0.1011  &
basis    0.3989  0.30078 0.21977 &
basis    0.6011  0.80078 0.28023 &
basis    0.8989  0.19922 0.78023 &
basis    0.1011  0.69922 0.71977 &
basis    0.6489  0.96977 0.44922 &
basis    0.8511  0.03023 0.94922 &
basis    0.1489  0.53023 0.55078 &
basis    0.3511  0.46977 0.05078 &
basis    0.46977 0.05078 0.3511  &
basis    0.53023 0.55078 0.1489  &
basis    0.03023 0.94922 0.8511  &
basis    0.96977 0.44922 0.6489  &
basis    0.55078 0.1489  0.53023 &
basis    0.05078 0.3511  0.46977 &
basis    0.44922 0.6489  0.96977 &
basis    0.94922 0.8511  0.03023 &
basis   0.125   0       0.25    &
basis   0.375   0       0.75    &
basis   0.875   0.5     0.25    &
basis   0.625   0.5     0.75    &
basis   0.25    0.125   0       &
basis   0.75    0.375   0       &
basis   0.25    0.875   0.5     &
basis   0.75    0.625   0.5     &
basis   0       0.25    0.125   &
basis   0       0.75    0.375   &
basis   0.5     0.25    0.875   &
basis   0.5     0.75    0.625   &
basis   0.875   0       0.75    &
basis   0.625   0       0.25    &
basis   0.125   0.5     0.75    &
basis   0.375   0.5     0.25    &
basis   0.75    0.875   0       &
basis   0.25    0.625   0       &
basis   0.75    0.125   0.5     &
basis   0.25    0.375   0.5     &
basis   0       0.75    0.875   &
basis   0       0.25    0.625   &
basis   0.5     0.75    0.125   &
basis   0.5     0.25    0.375   &

region wholething block 0 1 0 1 0 1 units lattice
create_box 4 wholething #no. is ntypes  ***

#after basis kw 1st no. is basis atm, 2nd is type
create_atoms    1       box     basis   1       1 #creates 160 atoms at
   once
#... will need to spec types
group Al1 id <> 1 16
group Al2 id <> 17 40
group Al union Al1 Al2
group O id <> 41 136
group Y id <> 137 160
set group Al type 1
set group O type 2
set group Y type 3



#Atom properties
mass 1 26.98 #Al
mass 2 15.999 #O
mass 3 88.906 #Y
mass 4 69.7 #Dopant Ga for Al

#can add a dopant as mass 4
set group O charge -2.0 #O (-2)
set group Al charge 3.0 #Al (+3) ...the doped for atom
set group Y charge 3.0 #Y (+3)

include runreqs.lmpin
```

```
fix relaxboxfix all box/relax iso $p
minimize 0 0 1000 1000

replicate $ud $ud $ud

#random doping
set group Al type/fraction 4 $dp $dseed  # ***
group Ga type 4
set group Ga charge 3.0 #Ga


#delete atoms
delete_atoms group Ga

variable T0 equal 1500 #i've got another T0
velocity all create ${T0} $vseed dist gaussian

reset_timestep 0
write_restart r.restart
```

Interatomic Potentials

runreqs.lmpin

```
timestep .002
kspace_style pppm .0001

pair_style hybrid/overlay born/coul/long 4.0 7.0 mbmh mbmh2
##2body params
#w/o Ga
pair_coeff 2 2 born/coul/long 2449.44 0.2907 0 0 0
pair_coeff 1 2 born/coul/long 1740.31 0.2907 0 0 0
pair_coeff 2 3 born/coul/long 1250.85 0.3497 0 0 0
pair_coeff 1 1 born/coul/long 312.11 0.071124 0 0 0
pair_coeff 3 3 born/coul/long 245.14 0.071124 0 0 0
pair_coeff 1 3 born/coul/long 256.55 0.071124 0 0 0
#w Ga, replaced the 1s w 4s  (but smaller no. has to be 1st
pair_coeff 2 4 born/coul/long 1740.31 0.2907 0 0 0
pair_coeff 4 4 born/coul/long 312.11 0.071124 0 0 0
pair_coeff 1 4 born/coul/long 312.11 0.071124 0 0 0
pair_coeff 3 4 born/coul/long 256.55 0.071124 0 0 0
##3body params
pair_coeff * * mbmh yag.mbmh A O Y A #maps 4th type to A
pair_coeff * * mbmh2 yag.mbmh2 A O Y A

#recovery from a binary w/ diffent no of procs
neigh_modify delay 0 every 1 check yes
```

3-Body parameters (yag.mbmh)

```
#3body only for certain kinds and only for XYY
#(a*(sigma=1)) is rij0 , lambda*(epsilon=1) is just lambda
# KEEP IT 1 FOR ALL EVEN IF NO THREE BODY INTERACTION!
```

```
#b/c same param used in 2body and we dont want to zero that

A A A 1.0   1.0   5.0   0.0   0.0   -0.333333333333 #cut@5
           312.11   14.06   3.0   3.0  0.0
A A O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
A A Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
A O A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
A O O 1.0   1.0   3.0   149.324   2.8   -0.333333333333# O-Al-O
           1740.31   3.44   -2.0   3.0  0.0
A O Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
A Y A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
A Y O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
A Y Y 1.0   1.0   5.0   0.0   0.0   -0.333333333333 #no 3b, cut@5
           256.55   14.06   3.0   3.0  0.0
O A A 1.0   1.0   3.0   0.0   0.0   -0.333333333333 #3b in 2nd file
           1740.31   3.44   -2.0   3.0  0.0
O A O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
O A Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #3b in 2nd file, no 2b
           0.0   0.0   0.0   0.0  0.0
O O A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
O O O 1.0   1.0   3.0   0.0   0.0   -0.333333333333 #cut@3
           2449.44   3.44   -2.0   -2.0  0.0
O O Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
O Y A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #3b in 2nd file, no 2b
           0.0   0.0   0.0   0.0  0.0
O Y O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
O Y Y 1.0   1.0   3.0   0.0   0.0   -0.333333333333 #3b in 2nd file
           1250.85   2.86   -2.0   3.0  0.0
Y A A 1.0   1.0   5.0   0.0   0.0   -0.333333333333 #no 3b, cut@5
           256.55   14.06   3.0   3.0  0.0
Y A O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
Y A Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
Y O A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
Y O O 1.0   1.0   3.0   168.25   2.8   -0.333333333333 #
           1250.85   2.86   3.0   -2.0  0.0
Y O Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
Y Y A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
Y Y O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
           0.0   0.0   0.0   0.0  0.0
```

```
Y Y Y 1.0   1.0   3.0   0.0   0.0   -0.333333333333 #no 3b, cut@3
          245.14   14.06   3.0   3.0  0.0
```

3-Body parameters (yag.mbmh2)

```
#3body only for certain kinds and only for XYY
#(a*(sigma=1)) is rij0, lambda*(epsilon=1) is just lambda
# KEEP IT 1 FOR ALL EVEN IF NO THREE BODY INTERACTION!
#actually zero the two-body interaction here

A A A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A A O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A A Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A O A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A O O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A O Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A Y A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A Y O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
A Y Y 1.0   1.0   5.0   0.0   0.0   -0.333333333333 #no 3b, cut@5
          256.55   14.06   3.0   3.0  0.0
O A A 1.0   1.0   2.6   6.242   2.0   -0.333333333333 #3b in this file
          0.0   0.0   0.0   0.0  0.0
O A O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O A Y 1.0   1.0   2.6   6.242   2.0   -0.333333333333 #3b in this file (symm
    ?), no 2b
          0.0   0.0   0.0   0.0  0.0
O O A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O O O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O O Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O Y A 1.0   1.0   2.6   6.242   2.0   -0.333333333333 #3b in this file (symm
    ?), no 2b
          0.0   0.0   0.0   0.0  0.0
O Y O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O Y Y 1.0   1.0   2.6   6.242   2.0   -0.333333333333 #3b in 2nd file, no 2b
          0.0   0.0   0.0   0.0  0.0
Y A A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y A O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y A Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
```

```
Y O A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y O O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y O Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y Y A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y Y O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y Y Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
```

Simulation Execution

```
##input
read_restart r.restart
#stuff not in restart file
include runreqs.lmpin

#1. input min struct 2. set T0 3. npt for a long time
#4. increase by 100 K or so until TF

variable udXts equal 100000 #10000*($ud)^3 #cook time
variable U equal etotal
variable PE equal pe
variable T equal temp
variable P equal press
variable a equal (vol^(1.0/3.0))/$ud
variable T0 equal 1500 #i've got another T0
variable ts equal step

group randa id 711 16 559 169 66 853 582 1126 844 871 &
143 607 894 621 825 289 178 76 233 1127

variable initloopno equal floor((${ts}/${udXts}))+1
variable todo equal 200-floor((${ts}/${udXts})) #currloop is floor+1
if ${todo}==0 then ”jump actions.lmpin bypass”

variable loopvar loop ${todo} #if set to 0 then will get an error
#that i don't care about

#want sim to go to right loop initially from read ts
#assign Ttgt, then continue to loop
label loop
variable currloopno equal ${loopvar}+${initloopno}-1 #a workaround for
    loopvar b/c starts w/ 1

variable Ttgt equal ${T0}+(${currloopno}-1)*10 #dT=10

velocity all scale ${Ttgt}
#use rescale for a step
#on 2nd thought i don't want abrupt v deltas
```

```
fix nptfix all npt temp ${Ttgt} ${Ttgt} 1 iso $p $p 10 drag 1
fix avgfix all ave/time 10 100 1000 v_U v_T v_P v_a v_PE file ${Ttgt}.${
    ts}.thermoavgs.avetime

if ”1500<=${Ttgt} && ${Ttgt}<=3500” then &
”jump actions.lmpin dumpandrdf ”
label dumpandrdf
dump rijdump randa custom 5 ${ts}.${Ttgt}.dump id type xu yu zu
compute rdfc all rdf 100
fix rdfout all ave/time 10 100 1000 c_rdfc file ${ts}.${Ttgt}.rdf mode
    vector

variable runfor equal ${udXts}*(${currloopno}) #has to be bigger than
    1000
run ${runfor} upto every 1000 ”write_restart r.restart”
if ”1500<=${Ttgt} && ${Ttgt}<=3500” then ”jump actions.lmpin postproc” &
else ”jump actions.lmpin dontpostproc”
label postproc
undump rijdump
uncompute rdfc
shell /home/aldosams/progs/sage/sage −python ../dumps2lis.py
label dontpostproc

next loopvar
jump actions.lmpin loop

label bypass
```

LAMMPS NON-EQUILIBRIUM THERMAL CONDUCTIVITY CODE

Simulation Setup Script

This script sets up the simulation.

```
label inits
dimension 3
units metal
boundary p p p
newton on
atom_style charge

variable a equal (${Tk}*9.67114349e-05)+12.3918
#geometry
#primitive vecs  a1 -.5 .5 .5 a2 .5 -.5 .5 a3 .5 .5 -.5 &
#conventional a1 1 0 0 a2 0 1 0 a3 0 0 1 &
#basis vecs below ,A1(16),A2(24),O(96),Y(24) in order, for conventional
#ERROR Input line too long!! changed input.cpp MAXLINES
lattice custom $a &
basis   0.0     0.0     0.0           &
basis   0.5     0       0.5     &
basis   0       0.5     0.5     &
basis   0.5     0.5     0       &
basis   0.75    0.25    0.25    &
basis   0.75    0.75    0.75    &
basis   0.25    0.25    0.75    &
basis   0.25    0.75    0.25    &
basis   0.25    0.75    0.75    &
basis   0.25    0.25    0.25    &
basis   0.75    0.75    0.25    &
basis   0.75    0.25    0.75    &
basis   0.5     0.5     0.5     &
basis   0       0.5     0       &
basis   0.5     0       0       &
basis   0       0       0.5     &
basis   0.375   0       0.25    &
basis   0.125   0       0.75    &
basis   0.625   0.5     0.25    &
basis   0.875   0.5     0.75    &
basis   0.25    0.375   0       &
basis   0.75    0.125   0       &
basis   0.25    0.625   0.5     &
basis   0.75    0.875   0.5     &
basis   0       0.25    0.375   &
basis   0       0.75    0.125   &
basis   0.5     0.25    0.625   &
basis   0.5     0.75    0.875   &
basis   0.75    0.625   0       &
basis   0.75    0.375   0.5     &
basis   0.25    0.875   0       &
basis   0.25    0.125   0.5     &
basis   0.125   0.5     0.25    &
basis   0.875   0       0.25    &
basis   0.375   0.5     0.75    &
basis   0.625   0       0.75    &
basis   0        0.25     0.875    &
basis   0.5      0.25     0.125    &
basis   0        0.75     0.625    &
basis   0.5      0.75     0.375    &
basis   0.28023  0.1011   0.19922  &
basis   0.21977  0.8989   0.69922  &
basis   0.71977  0.6011   0.30078  &
basis   0.78023  0.3989   0.80078  &
basis   0.19922  0.28023  0.1011   &
basis   0.69922  0.21977  0.8989   &
basis   0.30078  0.71977  0.6011   &
basis   0.80078  0.78023  0.3989   &
basis   0.1011   0.19922  0.28023  &
basis   0.8989   0.69922  0.21977  &
basis   0.6011   0.30078  0.71977  &
basis   0.3989   0.80078  0.78023  &
basis   0.8511   0.53023  0.05078  &
basis   0.6489   0.46977  0.55078  &
basis   0.3511   0.96977  0.94922  &
basis   0.1489   0.03023  0.44922  &
basis   0.03023  0.44922  0.1489   &
basis   0.96977  0.94922  0.3511   &
basis   0.46977  0.55078  0.6489   &
basis   0.53023  0.05078  0.8511   &
basis   0.94922  0.3511   0.96977  &
basis   0.44922  0.1489   0.03023  &
basis   0.05078  0.8511   0.53023  &
basis   0.55078  0.6489   0.46977  &
basis   0.71977  0.8989   0.80078  &
basis   0.78023  0.1011   0.30078  &
basis   0.28023  0.3989   0.69922  &
basis   0.21977  0.6011   0.19922  &
basis   0.80078  0.71977  0.8989   &
basis   0.30078  0.78023  0.1011   &
basis   0.69922  0.28023  0.3989   &
basis   0.19922  0.21977  0.6011   &
basis   0.8989   0.80078  0.71977  &
basis   0.1011   0.30078  0.78023  &
basis   0.3989   0.69922  0.28023  &
basis   0.6011   0.19922  0.21977  &
basis   0.1489   0.46977  0.94922  &
basis   0.3511   0.53023  0.44922  &
basis   0.6489   0.03023  0.05078  &
basis   0.8511   0.96977  0.55078  &
basis   0.96977  0.55078  0.8511   &
basis   0.03023  0.05078  0.6489   &
basis   0.53023  0.44922  0.3511   &
basis   0.46977  0.94922  0.1489   &
basis   0.05078  0.6489   0.03023  &
basis   0.55078  0.8511   0.96977  &
basis   0.94922  0.1489   0.46977  &
basis   0.44922  0.3511   0.53023  &
basis   0.78023  0.6011   0.69922  &
basis   0.71977  0.3989   0.19922  &
basis   0.21977 0.1011  0.80078 &
basis   0.28023 0.8989  0.30078 &
basis   0.69922 0.78023 0.6011  &
basis   0.19922 0.71977 0.3989  &
basis   0.80078 0.21977 0.1011  &
basis   0.30078 0.28023 0.8989  &
basis   0.6011  0.69922 0.78023 &
basis   0.3989  0.19922 0.71977 &
basis   0.1011  0.80078 0.21977 &
basis   0.8989  0.30078 0.28023 &
basis   0.3511  0.03023 0.55078 &
basis   0.1489  0.96977 0.05078 &
basis   0.8511  0.46977 0.44922 &
basis   0.6489  0.53023 0.94922 &
basis   0.53023 0.94922 0.6489  &
basis   0.46977 0.44922 0.8511  &
basis   0.96977 0.05078 0.1489  &
basis   0.03023 0.55078 0.3511  &
basis   0.44922 0.8511  0.46977 &
basis   0.94922 0.6489  0.53023 &
basis   0.55078 0.3511  0.03023 &
basis   0.05078 0.1489  0.96977 &
basis   0.21977 0.3989  0.30078 &
basis   0.28023 0.6011  0.80078 &
basis   0.78023 0.8989  0.19922 &
basis   0.71977 0.1011  0.69922 &
basis   0.30078 0.21977 0.3989  &
basis   0.80078 0.28023 0.6011  &
basis   0.19922 0.78023 0.8989  &
basis   0.69922 0.71977 0.1011  &
basis   0.3989  0.30078 0.21977 &
basis   0.6011  0.80078 0.28023 &
basis   0.8989  0.19922 0.78023 &
basis   0.1011  0.69922 0.71977 &
basis   0.6489  0.96977 0.44922 &
basis   0.8511  0.03023 0.94922 &
basis   0.1489  0.53023 0.55078 &
basis   0.3511  0.46977 0.05078 &
basis   0.46977 0.05078 0.3511  &
basis   0.53023 0.55078 0.1489  &
basis   0.03023 0.94922 0.8511  &
basis   0.96977 0.44922 0.6489  &
basis   0.55078 0.1489  0.53023 &
basis   0.05078 0.3511  0.46977 &
basis   0.44922 0.6489  0.96977 &
basis   0.94922 0.8511  0.03023 &
basis   0.125   0       0.25    &
basis   0.375   0       0.75    &
basis   0.875   0.5     0.25    &
basis   0.625   0.5     0.75    &
basis   0.25    0.125   0       &
basis   0.75    0.375   0       &
basis   0.25    0.875   0.5     &
basis   0.75    0.625   0.5     &
basis   0       0.25    0.125   &
basis   0       0.75    0.375   &
basis   0.5     0.25    0.875   &
basis   0.5     0.75    0.625   &
basis   0.875   0       0.75    &
basis   0.625   0       0.25    &
basis   0.125   0.5     0.75    &
basis   0.375   0.5     0.25    &
basis   0.75    0.875   0       &
basis   0.25    0.625   0       &
basis   0.75    0.125   0.5     &
basis   0.25    0.375   0.5     &
basis   0       0.75    0.875   &
basis   0       0.25    0.625   &
basis   0.5     0.75    0.125   &
basis   0.5     0.25    0.375   &

region wholething block 0 1 0 1 0 1 units lattice
create_box 4 wholething #no. is ntypes  ***

#after basis kw 1st no. is basis atm, 2nd is type
create_atoms    1       box     basis   1       1 #creates 160 atoms at once
#...will need to spec types
group Al1 id <> 1 16
group Al2 id <> 17 40
group Al union Al1 Al2
group O id <> 41 136
group Y id <> 137 160
set group Al type 1
set group O type 2
set group Y type 3


#Atom properties
mass 1 26.98 #Al
mass 2 15.999 #O
mass 3 88.906 #Y
mass 4 69.7 #Dopant Ga for Al

#can add a dopant as mass 4
set group O charge -2.0 #O (-2)
set group Al charge 3.0 #Al (+3) ...the doped-for atom
set group Y charge 3.0 #Y (+3)

include runreqs.lmpin

#looking at one side of sim box
variable dx equal $dx
variable hsx equal $hsx #heater or sink width
variable handsw equal ${hsx}*2
variable ly equal $ly
variable lz equal $lz
variable halfsimboxwidth equal ${dx}+${handsw}
variable simboxwidth equal 2*${halfsimboxwidth}

#create slab to be extruded
replicate 1 $ly $lz
replicate ${simboxwidth} 1 1


#random substitution
set group Al type/fraction 4 $dp $dseed  # ***
group Ga type 4
set group Ga charge 3.0 #Ga

# |0-sink-n1-dx-n2-heater-n3-heater-n4-dx-n5-sink-n6|
variable partg index lsink ldx lheater rheater rdx rsink
variable partl index $hsx $dx $hsx $hsx $dx $hsx
variable slabi equal 160*$ly*$lz
variable startid equal 1
label extrude
variable endid equal ${startid}+${slabi}*${partl}-1
group ${partg} id <> ${startid} ${endid}
variable startid equal ${endid}+1
next partg
next partl
jump setup.lmpin extrude

group sinks union lsink rsink
group heaters union lheater rheater
group grads union ldx rdx


variable T equal $Tk
variable TC equal $Tk-$dT/2
velocity all create ${TC} $vseed dist gaussian
variable TH equal $Tk+$dT/2

variable dToverdx equal $dT/$dx
variable dT equal $dT
velocity sinks scale ${TC}
velocity heaters scale ${TH}

variable nGa equal count(Ga)

#left side
variable xmn equal bound(ldx,xmin)
variable xmx equal bound(ldx,xmax)
if ${nGa} == 0 then &
”variable groupi index Al O Y” &
”variable tpg index lAl lO lY” else &
”variable groupi index Al O Y Ga” &
”variable tpg index lAl lO lY lGa”
label tpl
group ${tpg} intersect ldx ${groupi}
variable m index 26.98 15.99 88.906 69.7
variable VEhi equal 1.579347774*($m*${dT})^(1/2)/$m #-1.579347774*($m
    *300)^(1/2)/$m
velocity ${tpg} ramp vy 0 ${VEhi} x ${xmn} ${xmx} sum yes units box
next tpg
next groupi
next m
jump setup.lmpin tpl
variable tpg delete
variable groupi delete
variable m delete
#right side
variable xmn equal bound(rdx,xmin)
variable xmx equal bound(rdx,xmax)
if ${nGa} == 0 then &
”variable groupi index Al O Y” &
”variable tpg index rAl rO rY” else &
”variable groupi index Al O Y Ga” &
”variable tpg index rAl rO rY rGa”
label tpr
group ${tpg} intersect rdx ${groupi}
variable m index 26.98 15.99 88.906 69.7
variable VEhi equal 1.579347774*($m*${dT})^(1/2)/$m#-1.579347774*($m
    *300)^(1/2)/$m
velocity ${tpg} ramp vy 0 ${VEhi} x ${xmx} ${xmn} sum yes units box
next tpg
next groupi
next m
jump setup.lmpin tpr
variable tpg delete
variable groupi delete
variable m delete

reset_timestep 0
write_restart r.restart
```

Interatomic Potentials

runreqs.lmpin

```
timestep .001
kspace_style pppm .00001 #usually at .0001 but need some speed, .00001
    conserves but slow

#potentials block
pair_style hybrid/overlay born/coul/long 4.0 7.0 mbmh mbmh2
##2body params
#w/o Ga
pair_coeff 2 2 born/coul/long 2449.44 0.2907 0 0 0
pair_coeff 1 2 born/coul/long 1740.31 0.2907 0 0 0
pair_coeff 2 3 born/coul/long 1250.85 0.3497 0 0 0
pair_coeff 1 1 born/coul/long 312.11 0.071124 0 0 0
pair_coeff 3 3 born/coul/long 245.14 0.071124 0 0 0
pair_coeff 1 3 born/coul/long 256.55 0.071124 0 0 0
#w Ga, replaced the 1s w 4s (but smaller no. has to be 1st
pair_coeff 2 4 born/coul/long 1740.31 0.2907 0 0 0
pair_coeff 4 4 born/coul/long 312.11 0.071124 0 0 0
pair_coeff 1 4 born/coul/long 312.11 0.071124 0 0 0
pair_coeff 3 4 born/coul/long 256.55 0.071124 0 0 0
##3body params
pair_coeff * * mbmh /yag/commoninput/potentials/0/yag.mbmh A O Y A #maps
    4th type to A
pair_coeff * * mbmh2 /yag/commoninput/potentials/0/yag.mbmh2 A O Y A
#recovery from a binary w/ different no of procs
neigh_modify delay 0 every 1 check yes
```

3-Body parameters (yag.mbmh)

```
#3body only for certain kinds and only for XYY
#(a*(sigma=1)) is rij0, lambda*(epsilon=1) is just lambda
# KEEP IT 1 FOR ALL EVEN IF NO THREE BODY INTERACTION!
#b/c same param used in 2body and we dont want to zero that

A A A 1.0  1.0  5.0  0.0  0.0  -0.333333333333 #cut@5
         312.11  14.06  3.0  3.0 0.0
A A O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
A A Y 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
A O A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
A O O 1.0  1.0  3.0  149.324  2.8  -0.333333333333# O-Al-O
         1740.31  3.44  -2.0  3.0 0.0
A O Y 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
A Y A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
A Y O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
A Y Y 1.0  1.0  5.0  0.0  0.0  -0.333333333333 #no 3b, cut@5
         256.55  14.06  3.0  3.0 0.0
O A A 1.0  1.0  3.0  0.0  0.0  -0.333333333333 #3b in 2nd file
         1740.31  3.44  -2.0  3.0 0.0
O A O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
O A Y 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #3b in 2nd file, no 2b
         0.0  0.0  0.0  0.0 0.0
O O A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
O O O 1.0  1.0  3.0  0.0  0.0  -0.333333333333 #cut@3
         2449.44  3.44  -2.0  -2.0 0.0
O O Y 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
O Y A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #3b in 2nd file, no 2b
         0.0  0.0  0.0  0.0 0.0
O Y O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
O Y Y 1.0  1.0  3.0  0.0  0.0  -0.333333333333 #3b in 2nd file
         1250.85  2.86  -2.0  3.0 0.0
```

```
Y A A 1.0  1.0  5.0  0.0  0.0  -0.333333333333 #no 3b, cut@5
         256.55  14.06  3.0  3.0 0.0
Y A O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
Y A Y 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
Y O A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
Y O O 1.0  1.0  3.0  168.25  2.8  -0.333333333333 #
         1250.85  2.86  3.0  -2.0 0.0
Y O Y 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
Y Y A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
Y Y O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
Y Y Y 1.0  1.0  3.0  0.0  0.0  -0.333333333333 #no 3b, cut@3
         245.14  14.06  3.0  3.0 0.0
```

3-Body parameters (yag.mbmh2)

```
#3body only for certain kinds and only for XYY
#(a*(sigma=1)) is rij0, lambda*(epsilon=1) is just lambda
# KEEP IT 1 FOR ALL EVEN IF NO THREE BODY INTERACTION!
#actually zero the two-body interaction here

A A A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
A A O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
A A Y 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
A O A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
A O O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
A O Y 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
A Y A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
A Y O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
A Y Y 1.0  1.0  5.0  0.0  0.0  -0.333333333333 #no 3b, cut@5
         256.55  14.06  3.0  3.0 0.0
O A A 1.0  1.0  2.6  6.242  2.0  -0.333333333333 #3b in this file
         0.0  0.0  0.0  0.0 0.0
O A O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
O A Y 1.0  1.0  2.6  6.242  2.0  -0.333333333333 #3b in this file (symm
    ?), no 2b
         0.0  0.0  0.0  0.0 0.0
O O A 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
         0.0  0.0  0.0  0.0 0.0
O O O 1.0  1.0  0.0  0.0  0.0  -0.333333333333 #no 3b, no 2b
```

```
          0.0   0.0   0.0   0.0  0.0
O O Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O Y A 1.0   1.0   2.6   6.242   2.0   -0.333333333333 #3b in this file (symm
    ?), no 2b
          0.0   0.0   0.0   0.0  0.0
O Y O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
O Y Y 1.0   1.0   2.6   6.242   2.0   -0.333333333333 #3b in 2nd file, no 2b
          0.0   0.0   0.0   0.0  0.0
Y A A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y A O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y A Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y O A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y O O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y O Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y Y A 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y Y O 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
Y Y Y 1.0   1.0   0.0   0.0   0.0   -0.333333333333 #no 3b, no 2b
          0.0   0.0   0.0   0.0  0.0
```

Simulation Execution

```
label start
##input
read_restart r.restart
#stuff not in restart file
include runreqs.lmpin

set group Al type 1
set group O type 2
set group Y type 3
set group Ga type 4

variable a equal (${Tk}*9.67114349e-05)+12.3918
lattice sc 1 spacing $a $a $a

variable simsize equal (2*$hsx+$dx)*2*$lz*$ly
variable sXts equal 1000*(${simsize}) #cook time
variable U equal etotal
#variable Ta equal temp
variable ts equal step
variable dx equal $dx
variable hsx equal $hsx #heater or sink width
variable handsw equal ${hsx}*2
variable ly equal $ly
variable lz equal $lz
variable halfsimboxwidth equal ${dx}+${handsw}
variable Tk equal $Tk
variable TH equal $Tk+$dT/2
variable TC equal $Tk-$dT/2
variable dToverdx equal $dT/$dx
variable dT equal $dT
variable dTH equal ${dT}/2
variable dTC equal -${dT}/2
compute kea all ke/atom
#converts KE to temp
variable Ta atom "7736.34*c_kea" #=kb/1.5=11604/1.5

#assignment of groups of atoms
# |0-sink-n1-dx-n2-heater-n3-heater-n4-dx-n5-sink-n6|
variable partg index lsink ldx lheater rheater rdx rsink
variable partl index $hsx $dx $hsx $hsx $dx $hsx
variable slabi equal 160*$ly*$lz
variable startid equal 1
label extrude
variable endid equal ${startid}+${slabi}*${partl}-1
group ${partg} id <> ${startid} ${endid}
variable startid equal ${endid}+1
variable xmn equal bound(${partg},xmin)
variable xmx equal bound(${partg},xmax)
print bounds for ${partg}: min: ${xmn}, max: ${xmx}
next partg
next partl
jump actions.lmpin extrude

group sinks union lsink rsink
group heaters union lheater rheater
group grads union ldx rdx


fix lgf ldx nve
fix rgf rdx nve
fix lheaterf lheater nvt temp ${TH} ${TH} 1
fix rheaterf rheater nvt temp ${TH} ${TH} 1
#why have to separate?!
fix lsinkf lsink nvt temp ${TC} ${TC} 1
fix rsinkf rsink nvt temp ${TC} ${TC} 1
fix_modify lheaterf energy yes
fix_modify rheaterf energy yes
fix_modify lsinkf energy yes
fix_modify rsinkf energy yes


#two stages: 1. thermalize 2. ss while recording
#what stage is the sim at?
if ${ts} < ${sXts} then "jump actions.lmpin transient" &
else "jump actions.lmpin ss"
```

```
label transient
run ${sXts} upto every 1000 ”write_restart r.restart”
#need to erase fix data b/c i don't want to record it
unfix lgf
unfix rgf
unfix lheaterf
unfix rheaterf
unfix lsinkf
unfix rsinkf
write_restart r.restart #so i don't save previous fix info

label ss
fix lgf ldx nve
fix rgf rdx nve
fix lheaterf lheater nvt temp ${TH} ${TH} 1
fix rheaterf rheater nvt temp ${TH} ${TH} 1
fix lsinkf lsink nvt temp ${TC} ${TC} 1
fix rsinkf rsink nvt temp ${TC} ${TC} 1
fix_modify lheaterf energy yes
fix_modify rheaterf energy yes
fix_modify lsinkf energy yes
fix_modify rsinkf energy yes
#energy that has gone thru sys (each side).. /4 is for avg. /2 to split
    sys E in 2
run 0
#variable Epc equal 100*(sqrt(f_lsinkf^2)+sqrt(f_lheaterf^2)+sqrt(
    f_rheaterf^2)+sqrt(f_rsinkf^2))/(4*sqrt(etotal^2)/2)
#just so i can have file names w/ timesteps
fix avgfix all ave/spatial 10 100 1000 x lower 1 v_Ta file ${ts}.
    tempprofile.avespacial
fix qf all ave/time 1 1 1000 f_lsinkf f_lheaterf f_rheaterf f_rsinkf
    file ${ts}.cumdE.avetime
if ${ts} > 1000000 then ”jump actions.lmpin end”
label keepgoing
run 1000
write_restart r.restart
run 0 #idk why i need to include this
#if ${Epc} > .1 then ”jump actions.lmpin end” else ”jump actions.lmpin
    keepgoing”
if ${ts} > 1000000 then ”jump actions.lmpin end” else ”jump actions.
    lmpin keepgoing”

label end
```

BIBLIOGRAPHY

[1] “Jmol: an open-source java viewer for chemical structures in 3d,” On the WWW, URL `http://www.jmol.org/`.

[2] L. Dobrzycki, E. Bulska, D. A. Pawlak, Z. Frukacz, and K. Wozniak, “Structure of YAG crystals doped/substituted with erbium and ytterbium,” *Inorganic Chemistry*, vol. 43, no. 24, pp. 7656–7664, 2004, pMID: 15554630.

[3] A. Potdevin, G. Chadeyron, and R. Mahiou, “Tb^{3+}-doped yttrium garnets: Promising tunable green phosphors for solid-state lighting,” *Chemical Physics Letters*, vol. 490, no. 1-3, pp. 50 – 53, 2010.

[4] R. Hansel, S. Allison, and G. Walker, “Temperature-dependent luminescence of gallium-substituted YAG:Ce,” *Journal of Materials Science*, vol. 45, pp. 146–150, 2010, 10.1007/s10853-009-3906-9.

[5] X. Q. Cao, R. Vassen, and D. Stoever, “Ceramic materials for thermal barrier coatings,” *Journal of the European Ceramic Society*, vol. 24, no. 1, pp. 1 – 10, 2004.

[6] A. Ikesue, Y. L. Aung, T. Taira, T. Kamimura, K. Yoshida, and G. L. Messing, “Progress in ceramic lasers,” *Annual Review of Materials Research*, vol. 36, no. 1, pp. 397–429, 2006.

[7] K. Sattler, *Handbook of Nanophysics: Nanoparticles and Quantum Dots*, ser. Handbook of Nanophysics.

[8] J. L. Caslavsky and D. J. Viechnicki, “Melting behaviour and metastability of yttrium aluminium garnet (YAG) and $YAlO_3$ determined by optical differential thermal analysis,” *Journal of Materials Science*, vol. 15, pp. 1709–1718, 1980, 10.1007/BF00550589.

[9] R. Wynne, J. L. Daneu, and T. Y. Fan, “Thermal coefficients of the expansion and refractive index in YAG,” *Appl. Opt.*, vol. 38, no. 15, pp. 3282–3284, 1999.

[10] Red Optronics, “Undoped yag crystal ($Y_3Al_5O_{12}$),” On the WWW, URL `http://www.redoptronics.com/undoped-YAG-crystal.html`.

[11] H. Qiu, P. Yang, J. Dong, P. Deng, J. Xu, and W. Chen, “The influence of Yb concentration on laser crystal Yb:YAG,” *Materials Letters*, vol. 55, no. 1-2, pp. 1 – 7, 2002.

[12] J. Morikawa, C. Leong, T. Hashimoto, T. Ogawa, Y. Urata, S. Wada, M. Higuchi, and J. ichi Takahashi, “Thermal conductivity/diffusivity of Nd^{3+} doped $GdVO_4$, YVO_4, $LuVO_4$, and $Y_3Al_5O_{12}$ by temperature wave analysis,” *Journal of Applied Physics*, vol. 103, no. 6, p. 063522, 2008.

[13] J. F. W. Gunsser, U. Wolfmeier, *1.3.8.4 Debye Temperature*, A. M. H. K.-H. Hellwege, Ed. Springer-Verlag, 1978, vol. 12a.

[14] J. Bodzenta, A. Kaźmierczak-Bałata, K. B. Wokulska, J. Kucytowski, T. Łukasiewicz, and W. Hofman, "Influence of doping on thermal diffusivity of single crystals used in photonics: measurements based on thermal wave methods," *Appl. Opt.*, vol. 48, no. 7, pp. C46–C54, 2009.

[15] C. Jun, C. Dong-Quan, and Z. Jing-Lin, "Molecular dynamics simulation of thermodynamic properties of YAG," *Chinese Physics*, vol. 16, no. 9, p. 2779, 2007.

[16] S. Plimpton, "Fast parallel algorithms for short-range molecular-dynamics," *Journal of Computational Physics*, vol. 117, no. 1, pp. 1–19, March 1 1995.

[17] L. Verlet, "Computer "Experiments" on Classical Fluids. II. Equilibrium Correlation Functions," *Phys. Rev.*, vol. 165, no. 1, pp. 201–214, Jan 1968.

[18] M. P. Tosi and F. G. Fumi, "Ionic sizes and born repulsive parameters in the NaCl-type alkali halides–II : The generalized Huggins-Mayer form," *Journal of Physics and Chemistry of Solids*, vol. 25, no. 1, pp. 45 – 52, 1964.

[19] *Z. Phys.*, vol. 11.

[20] M. E. Glicksman, *Principles of Solidification : an introduction to modern casting and crystal growth concepts*. New York, New York: Springer, 2011, ch. 1, p. 17.

[21] J. G. O. Ojwang, R. van Santen, G. J. Kramer, A. C. T. van Duin, and I. W. A. Goddard, "Predictions of melting, crystallization, and local atomic arrangements of aluminum clusters using a reactive force field," vol. 129, no. 24, p. 244506, 2008.

[22] D. P. Sellan, E. S. Landry, J. E. Turney, A. J. H. McGaughey, and C. H. Amon, *Phys. Rev. B*, vol. 81, Jun.