Online simulations via nanoHUB: Glass transition temperature of an amorphous polymer

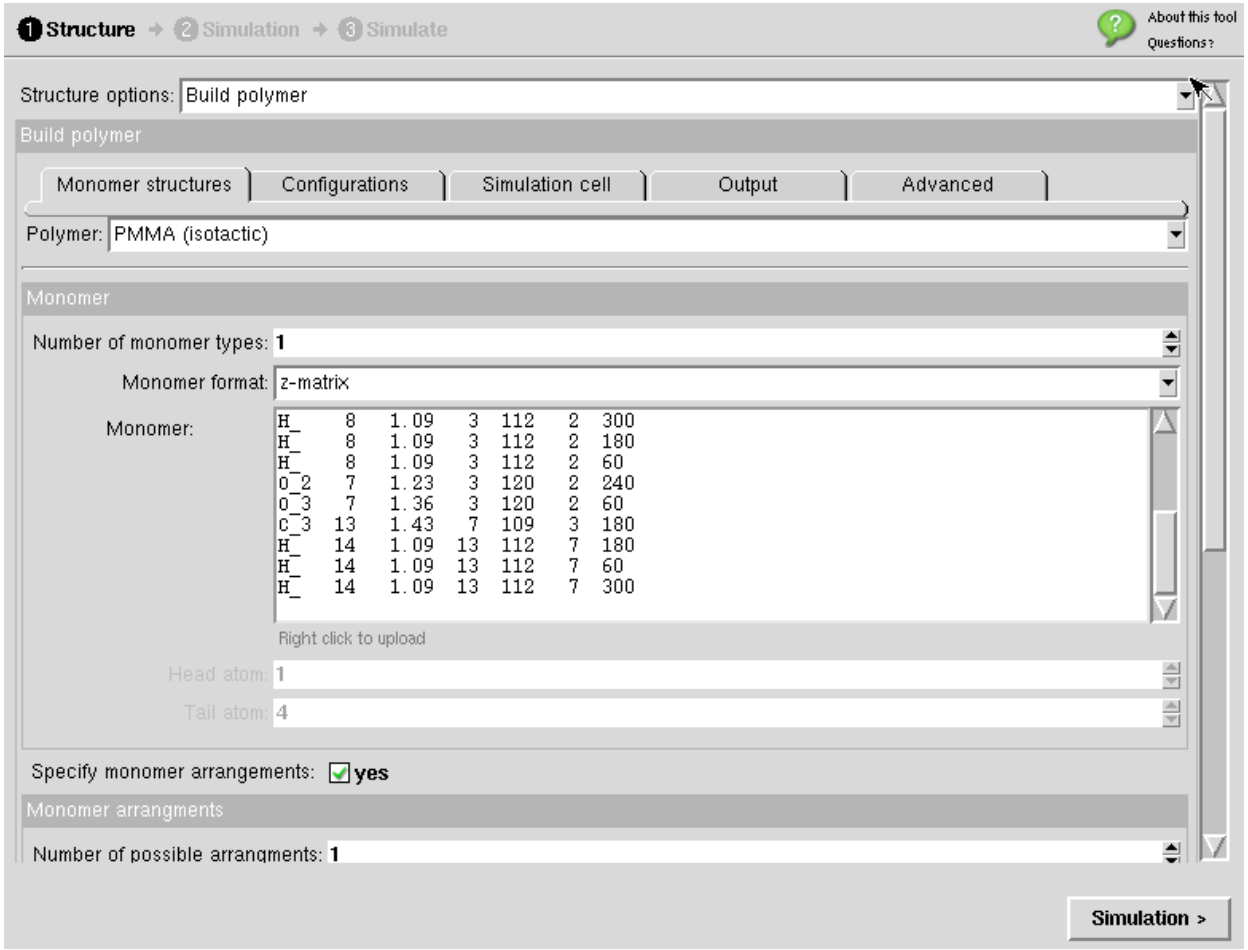
In this tutorial:

- Build and atomic structure of an amorphous polymer
- Perform a cooling simulation using MD
- Characterize its glass transition temperature

David Lowing, Ben Haley, Ale Strachan
strachan@purdue.edu
School of Materials Engineering & Birck Nanotechnology Center
Purdue University
West Lafayette, Indiana USA

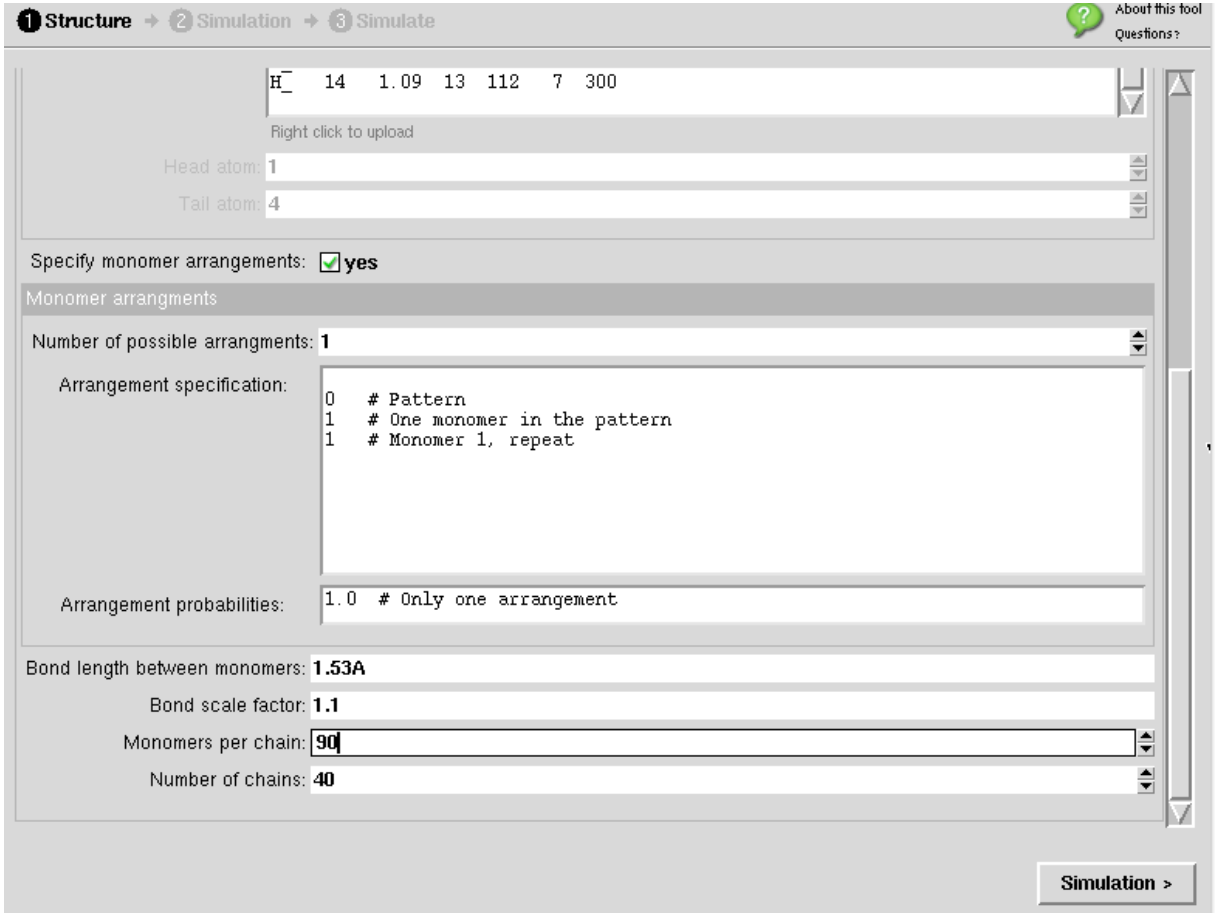
STEP 1: Launch PolymerModeler

- Launch PolymerModeler, and you will see the image below
- Make sure PMMA (isotactic) is selected as the polymer to be used



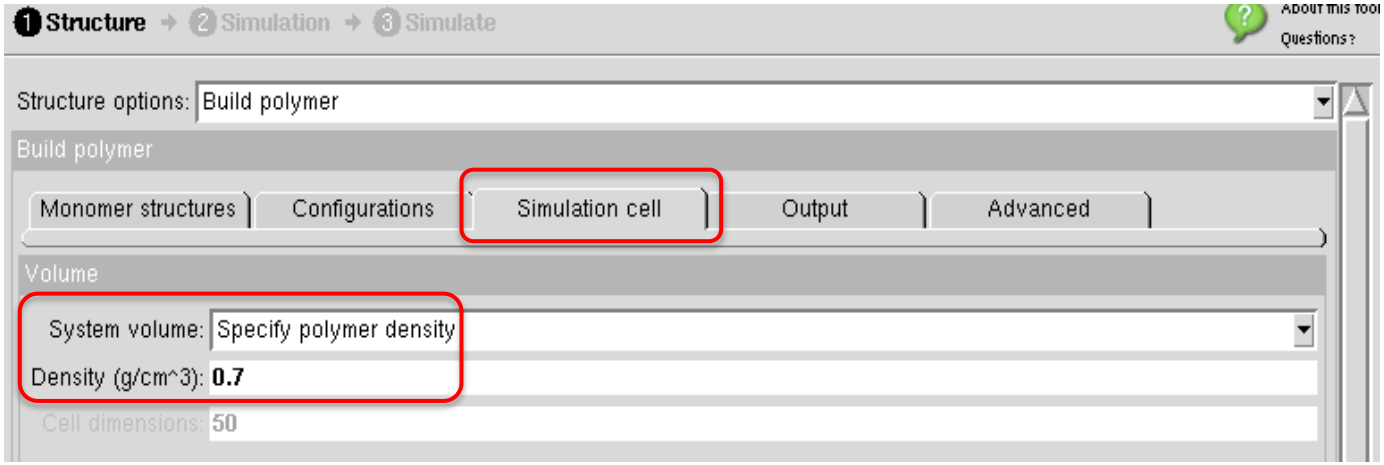
STEP 2: Polymer Chain Construction

- With the “Monomer structures” tab selected scroll down to the bottom
- Set “Monomers per chain” and “Number of chains” as desired; for this example we will use 40 chains with 90 monomers each



STEP 3: Density Selection

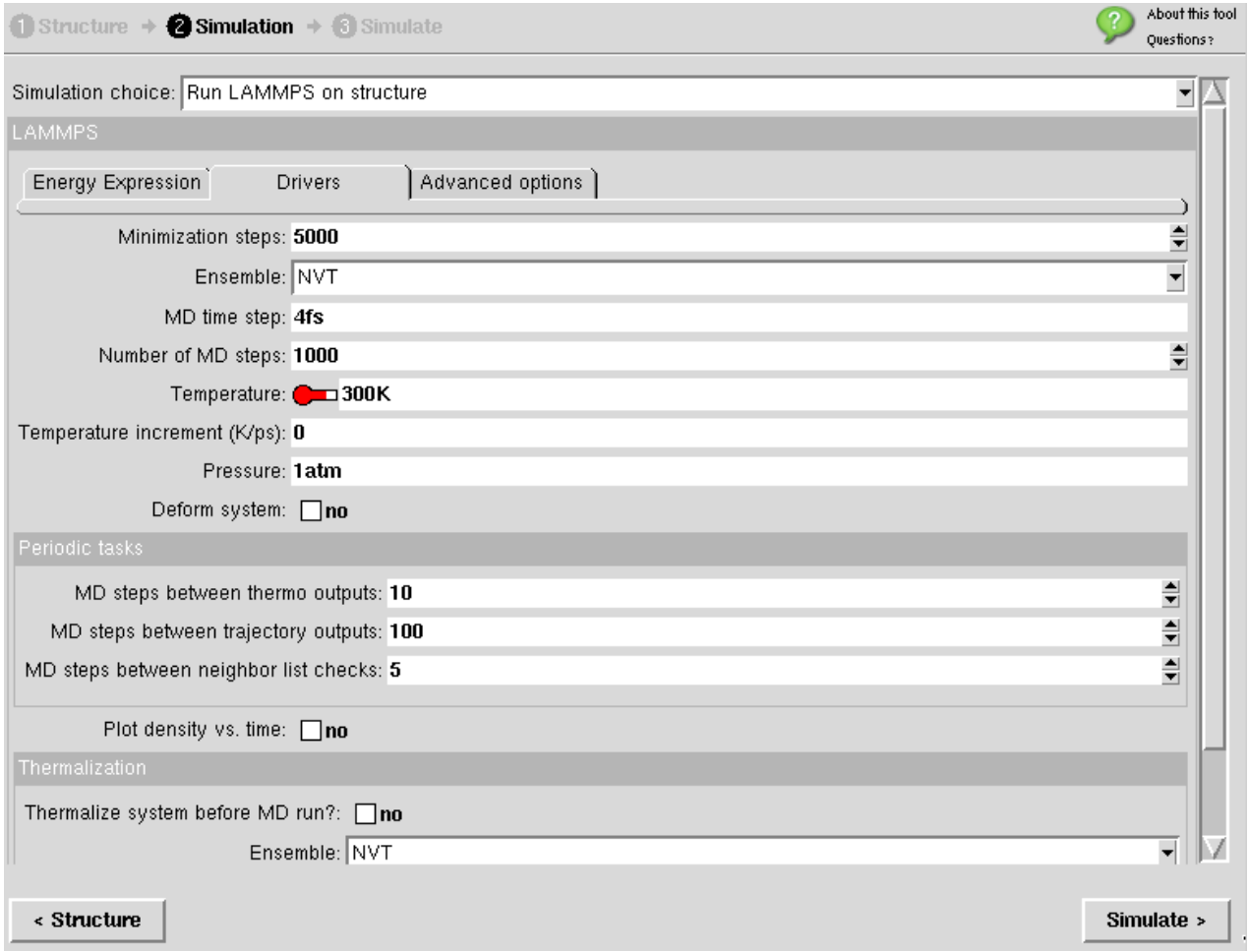
- Select the “Simulation cell” tab
- Under “System volume” select “Specify polymer density” and then choose 0.7 g/cm^3



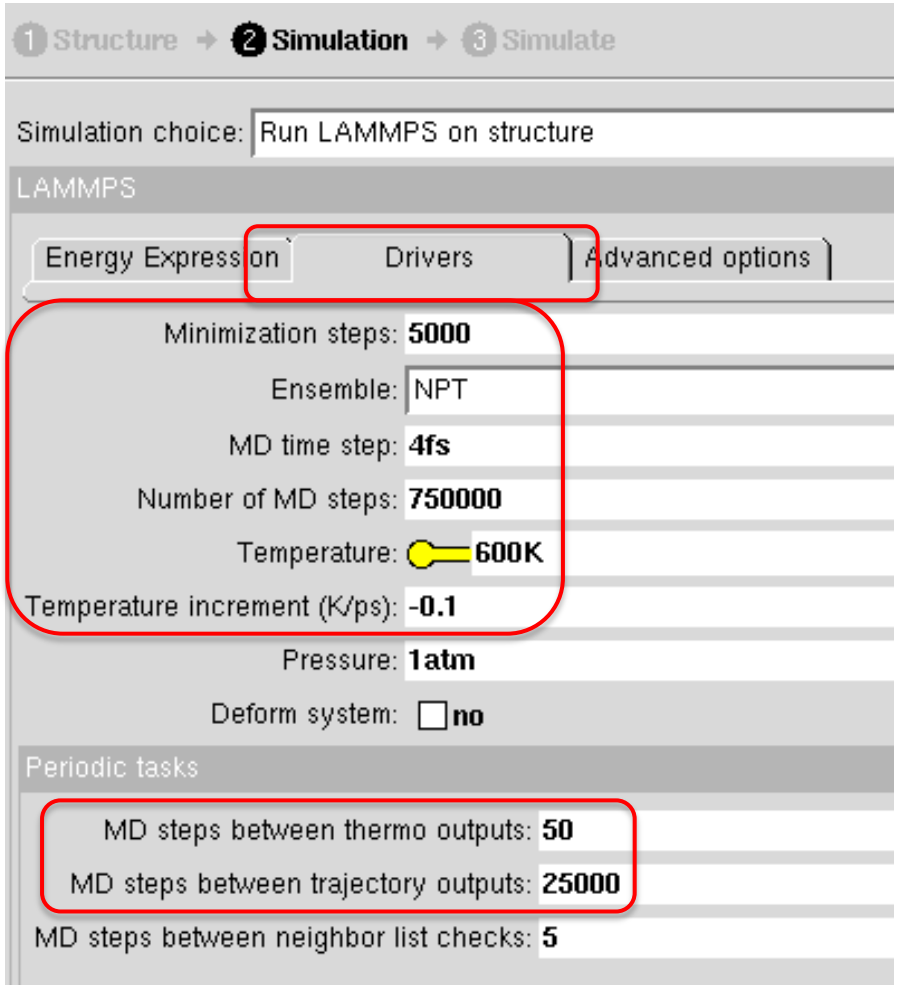
- Now click on the simulation button at the bottom right hand corner of the tool to proceed and select simulation parameters

STEP 4: Simulation Parameters Part 1

- In the “Simulation” phase choose the “Drivers” tab
- Here you design the test that will be run on your polymer by adjusting many different conditions



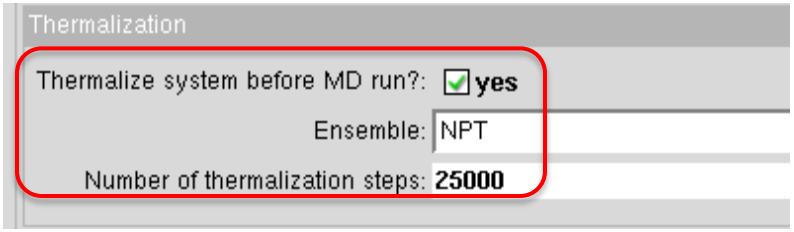
STEP 4: Simulation Parameters Part 2



- For this example we will use the following conditions:
  - Temperature 600K and temperature increment -0.1 K/ps (cool your system to 300K over a 3000 ps)
  - Ensemble NPT and keep the minimization steps at 5,000
  - Set the number of MD steps to 750,000
  - The outputs that you receive and how frequent you receive them are set in the periodic task section

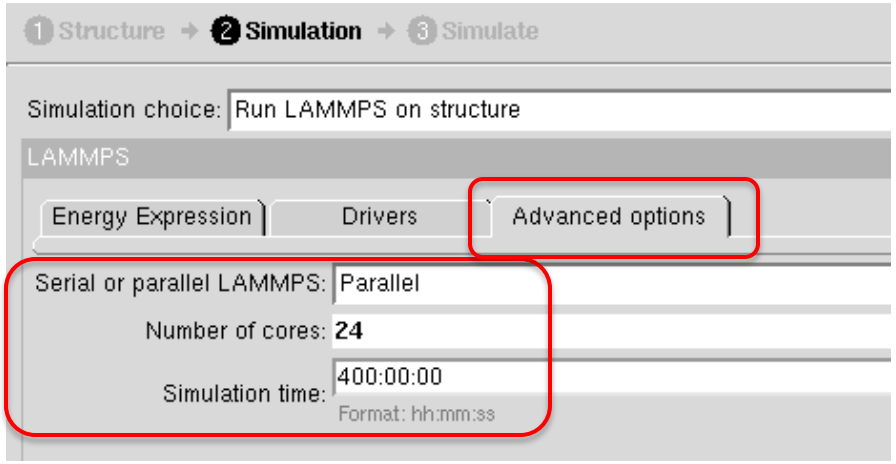
STEP 5: Thermalization

- Scroll down to the bottom of the “Drivers” tab.
- For this example we want to thermalize the polymer before cooling the system.
- Set the thermalization ensemble to NPT and the number of steps to 25,000.

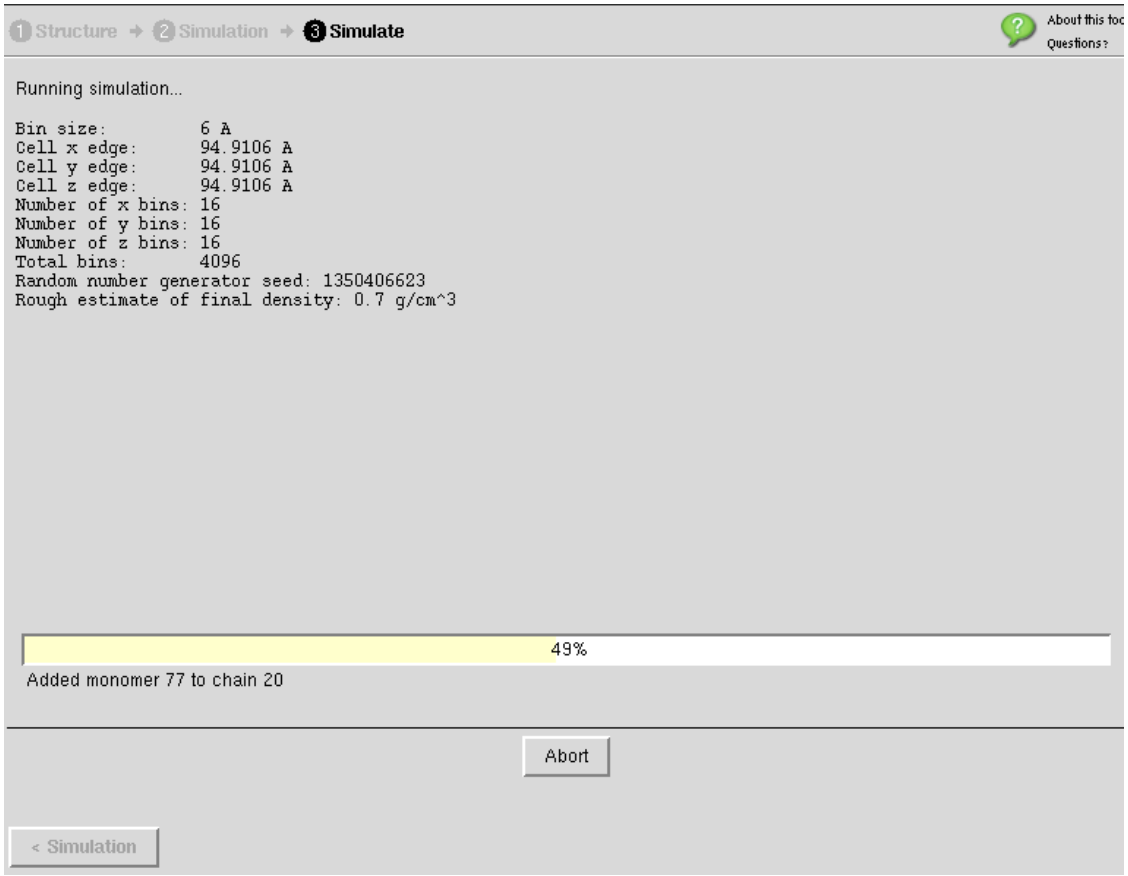


STEP 6: nanoHUB Computing Power

- Now select the “Advanced options” tab
- Here you can select how much computing power nanoHUB will use to run the simulation
- For this calculation, due to its size, you want to run parallel LAMMPS
- Select 24 cores and set the time at 400 hours; this will be more than enough processing power and time for the simulation to be completed
- Finally press the “Simulate” button in the bottom right hand corner to run the simulation

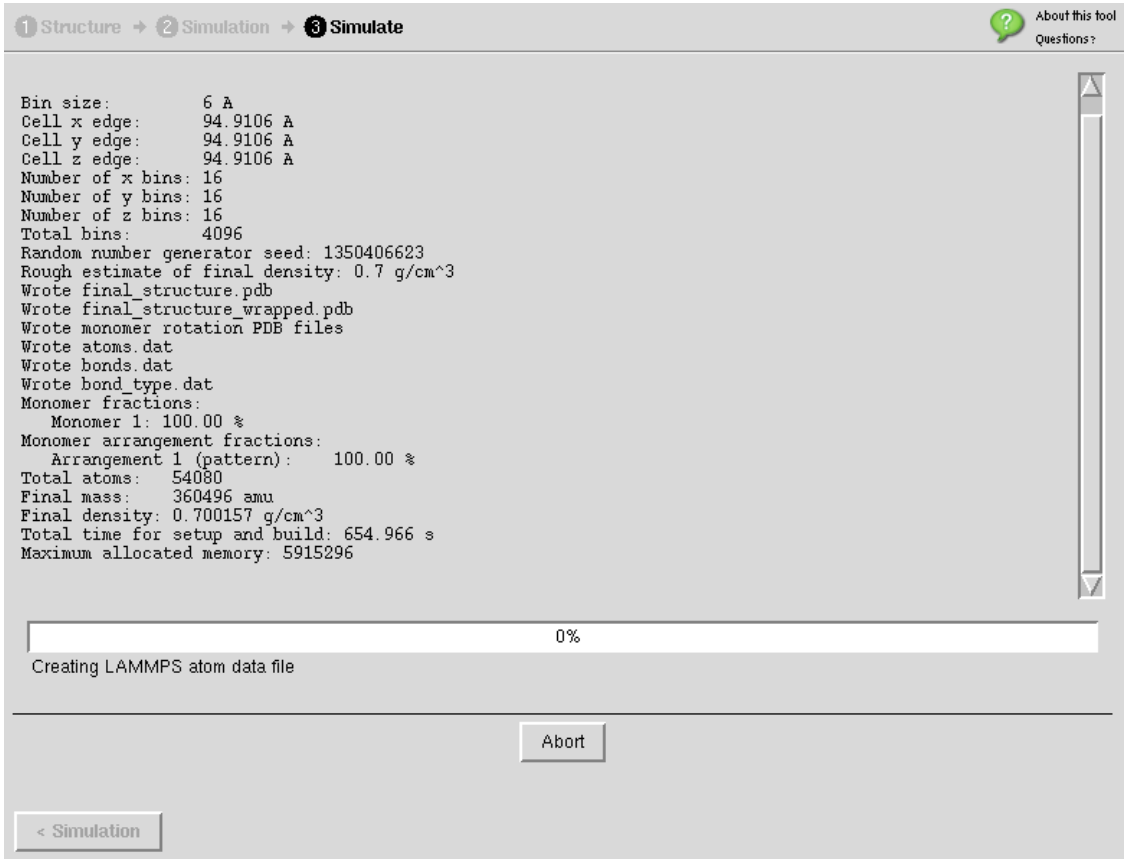


STEP 7: Building The Polymer



- When the simulation runs you will first see this screen, indicating that PolymerModeler is building the polymer chains.
- Once the chains are built, PolymerModeler will automatically run LAMMPS.

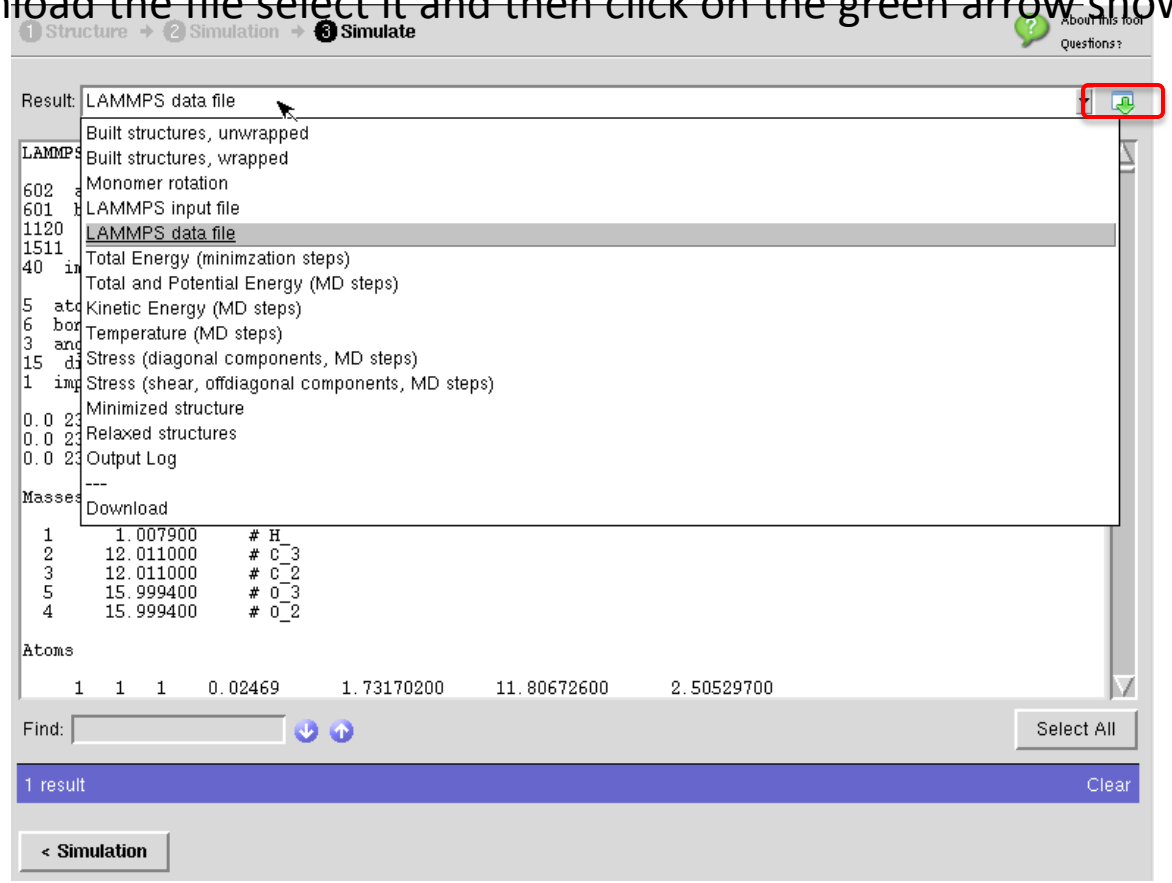
STEP 8: Running the Simulation



- You won't see any new output while parallel LAMMPS is running. LAMMPS runs on a cluster machine outside nanoHUB.
- This stage will take some time to complete

STEP 9: Outputs

- When the simulation is complete you will see a screen with the outputs
- You will need to download the Output Log at the bottom of the drop down list
- This file will contain all the data required to calculate the Tg of the polymer
- To download the file select it and then click on the green arrow shown below



STEP 10: Data Extraction

- To find the Tg we need to plot the density as a function of temperature so we need to extract the temperature and volume data from the output log.
- The output log is very large, but the data is split into sections based on the stage of the simulation (minimization, thermalization, cooling).
- Scroll down though the output log until you reach the cooling stage section
  - The third section is easy to find; just scroll down until you reach a step number greater than 50,000 (for this example).
  - Scroll up until the step number is 0 and that will be beginning of the cooling section
  - There will be data above it as it is not the top of the output log
- The temperature data is the 4th column and the x, y, z volume components are the last 3 columns respectively (example values are shown below)
- You need to extract these four (entire) columns.

Temperature

```
598.45079
 598.4652
595.91793
598.07177
598.97102
599.05297
601.86532
 601.2352
603.36396
603.96311
602.60066
```

x, y, z components

91.097052	91.097052	91.097052
91.091029	91.091029	91.091029
91.076681	91.076681	91.076681
91.053639	91.053639	91.053639
91.025385	91.025385	91.025385
90.990958	90.990958	90.990958
90.953234	90.953234	90.953234
90.911455	90.911455	90.911455
90.866827	90.866827	90.866827
90.825324	90.825324	90.825324

STEP 11: Mass and Density Calculations

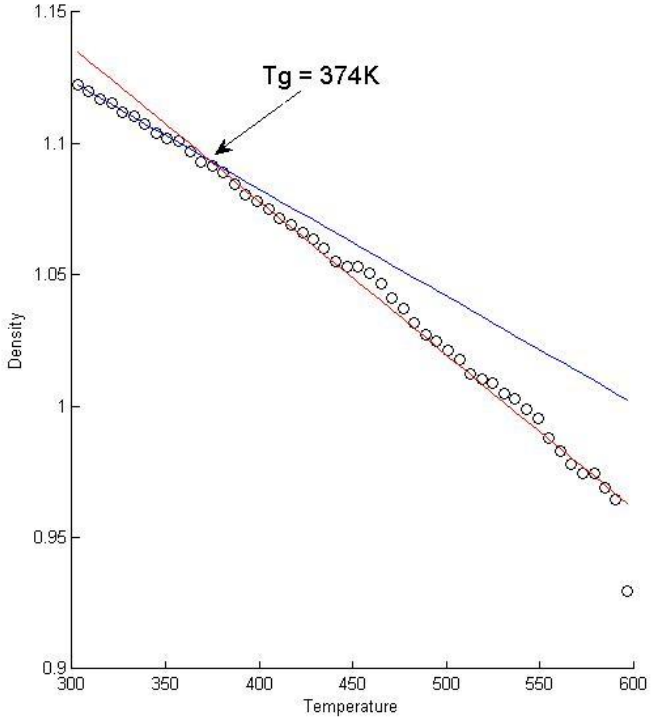
- The next step is to calculate the mass of the polymer
  - Take the initial x, y, z volume components and multiply them together to get the iniital volume.
  - We set the initial volume to be 0.7 g/cm^3 but the data is in terms of amu/angstrom^3 so we divide 0.7 by a conversion factor of 1.66053886.
  - We then multiply our initial density by the initial volume to get the mass.
- The mass of the sample is constant, but the density and volume will change throughout the simulation; we need to apply this procedure to each step of the simulation.
  - For each step you divide the mass that you calculated above by the volume at each step (V = Vx*Vy*Vz)
  - As a result, you have a new column of density values.

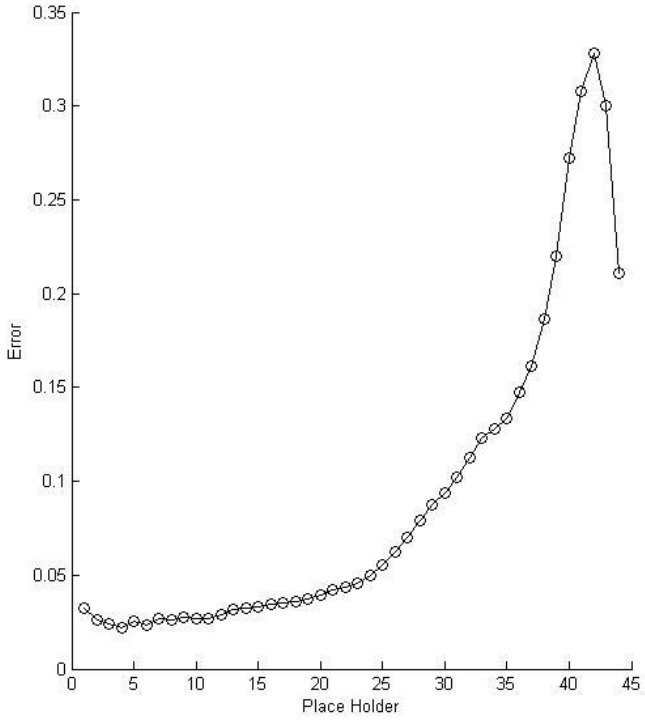
STEP 12: Plotting Density vs. Temperature

- Before making the plot you need to reduce the number of data values you have for density and temperature.
- Do this by breaking your data up into 50 different sections and taking an average density and temperature value for each.
- Plot the 50 average densities vs. temperatures; the glass transition temperature (Tg) should be visible as the relative point where the slope of the density vs. T curve changes.
- To computationally find the temperature where the slope changes on your graph you must perform a double linear regression on your plot of density as a function of temperature. This process fits two straight lines to the parts of the curve above and below Tg; the point at which the lines cross is Tg.
- This can be carried out in different computational packages such as MATLAB or MATHCAD, as shown on the next slide.

STEP 13: Results Part 1

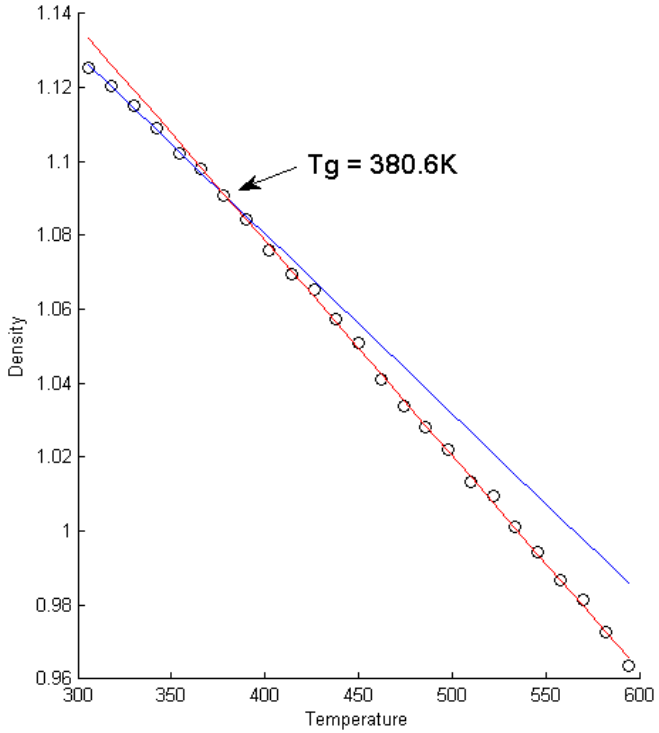
- The plot below shows the results of the double linear regression with Tg = 374K.
- The error graph shows the error of the different sets of line combinations that the regression program calculated; the lowest error is used to determine the final Tg value.

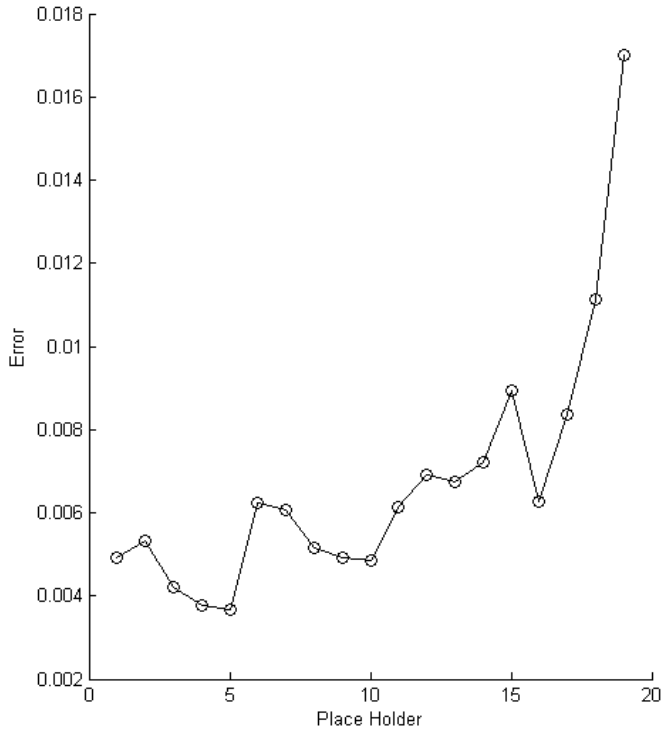




STEP 14: Results Part 2

- Now we have results for one simulation, but to confirm our results we should run the simulation again to confirm no errors occurred.
- Shown below are the results of running the same simulation again.
- The resulting Tg = 380.6K.
- Results reported in literature include 388K [1] and 381K [2]





[1] T. G. Fox, B. S. Garrett, W. E. Goode, Serge Gratch, J. F. Kincaid, Aldenlee Spell, and J. D. Stroupe *Journal of the American Chemical Society* 1958

[2] Chengzhi, C.; Kristoffer, A.; Jørgen, L. J. J. Appl. Polym. Sci. 2004