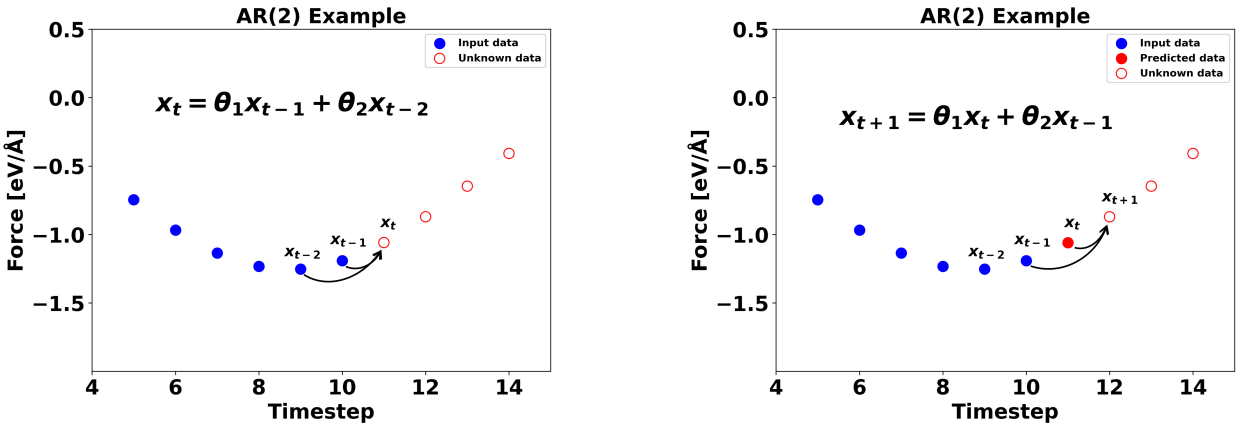
Master thesis outline

Birk Dissing

August 2024

1 Introduction

- Molecular dynamics is an important tool in modern chemistry as it allows chemists to explore chemical phenomena at a much smaller physical and temporal scale than otherwise possible. It also allows for the exploration of molecules that would otherwise be difficult to work with.
- However the small timescale of molecular simulation can also be its biggest weakness as it makes it nigh on impossible to investigate rare events or events that take a "long" time. (Talk about limits of molecular dynamics)
- An example of where this becomes a problem is a material growth, which suffers both from rare events, nucleation events, as well as interest in a longer timescale, the growth phase. The exact mechanics of material growth is currently unknown and needs information with greater temporal resolution than experiments yield. Focus has therefore turned to how to make rare events more common in molecular dynamics as well as how to speed it up.
  - Speeding up molecular dynamics will affect all fields. If a way to speed up the simulations with limited reduction in accuracy were found, data would be acquired faster for all scientists doing simulations.
- There are different ways of making rare events less rare and increasing the speed of simulations. For example, one could take a trajectory where you know a nucleation event occurs and run a simulation where you alter it slightly. There is also work being done on several ways to speed up simulations, one way is to create machine learning(ML) force fields, which can replace the computationally expensive force calculations. (Other methods for nucleation events) [1]
- A problem with ML force fields is that while they give speed-ups, they are often specialized to specific molecular species and require a lot of work and data to train the force field. This means that the large rewards of using ML fields come at a large investment cost. Our goal is to create a method that can speed up simulations without requiring a large amount of training. We want to do this by taking the forces calculated for the last couple of timesteps and using them to predict future forces. While this model will most likely lead to a smaller speed-up than an optimized ML model the model will hopefully be generally applicable without any extra setup for many different chemical systems.



(a) (b)

Figure 1: The black dots are data in a force trajectory while the red dots are values predicted by an $AR(2)$ model. In a) only data originally in the time series is used to predict the next point. In b) a point previously predicted by the model is used to predict the next point.

2 ARIMA

2.1 AR models

- There are different ways of predicting points in a time series you could do that by intuition or using equations like Newton's laws. However, when you want to predict the next points in a more complicated time series you would use an auto-regressive (AR) model which creates a model that can best predict the training data. The formula for an AR model is

$$AR(p) : x_t = \sum_{i=1}^{p} \theta_i x_{t-i}, \tag{1}$$

where x_t are points in the time series X, θ are fitted parameters for the model, and p is the lag which denotes the amount of previous values in the time series used to predict future values. An example of how an AR model works can be seen in figure 1.

- AR models only use information from their time series to predict future values. However, there is often information in other time series that can be used to enhance the prediction. For example, pressure, temperature, and cloud cover all hold information useful in predicting the chance of rain. For example, if we want to use information in time series Y to predict time series X using two points it would look like this:

$$x_t = \theta_{1,x} x_{t-1} + \theta_{2,x} x_{t-2} + \theta_{3,x} y_{t-1} + \theta_{4,x} y_{t-2}. \tag{2}$$

In this case, we also need a model predicting Y:

$$y_t = \theta_{1,y} x_{t-1} + \theta_{2,y} x_{t-2} + \theta_{3,y} y_{t-1} + \theta_{4,y} y_{t-2} \tag{3}$$

- We decided to use a VAR model using all of the force trajectories of a molecule. As we would need to predict all of the forces anyway creating one single model that can utilize information on all force trajectories while predicting seems better than many

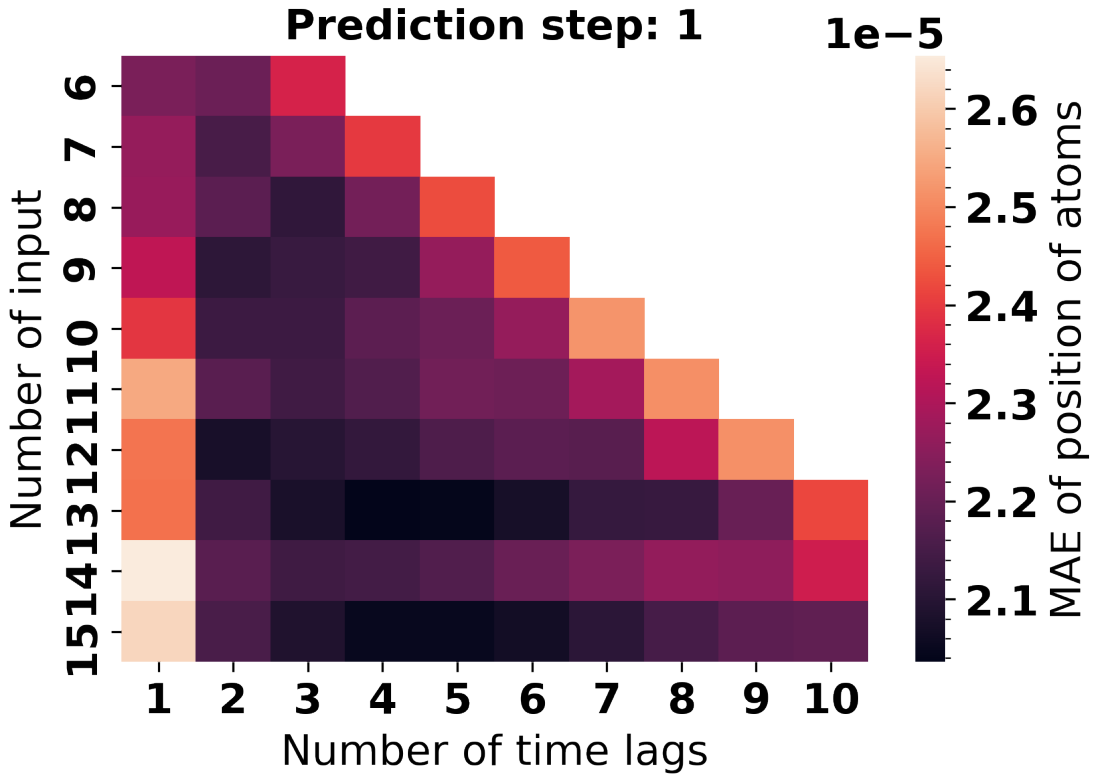


Figure 2: The figure shows the MAE of models with the given values for input and order. **Det bør være MAE af forces for det er lettere at forstå.**

models each using and predicting only a single trajectory. In order to fit the model we create a linear least square problem that can be solved using SVD.

$$endog = \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \\ x_4 & y_4 \\ x_5 & y_5 \end{bmatrix}, \quad z = \begin{bmatrix} x_2 & y_2 & x_1 & y_1 \\ x_3 & y_3 & x_2 & y_2 \\ x_4 & y_4 & x_3 & y_3 \end{bmatrix}, \quad y = \begin{bmatrix} x_3 & y_3 \\ x_4 & y_4 \\ x_5 & y_5 \end{bmatrix} \tag{4}$$

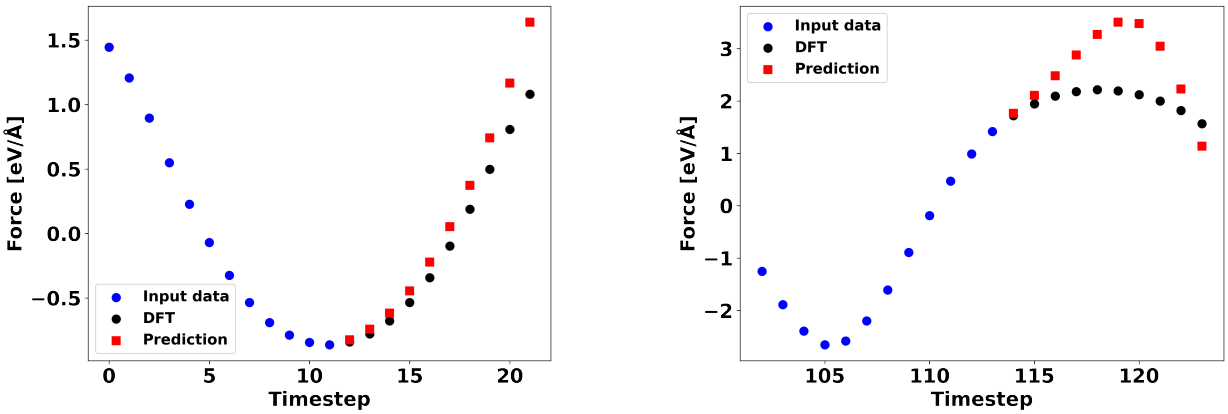
$$z\Theta = y \Rightarrow \begin{bmatrix} x_2 & y_2 & x_1 & y_1 \\ x_3 & y_3 & x_2 & y_2 \\ x_4 & y_4 & x_3 & y_3 \end{bmatrix} \begin{bmatrix} \theta_{1,x} & \theta_{1,y} \\ \theta_{2,x} & \theta_{2,y} \\ \theta_{3,x} & \theta_{3,y} \\ \theta_{4,x} & \theta_{4,y} \end{bmatrix} = \begin{bmatrix} x_3 & y_3 \\ x_4 & y_4 \\ x_5 & y_5 \end{bmatrix} \tag{5}$$

- In order to optimize the parameters of the model a grid search was performed on the number of inputs used to fit the model and the order p of the VAR model. The models in the grid search only make 1 prediction as the error increases exponentially and it was decided to first test a working model that makes 1 prediction. The result of the grid search can be seen in figure 2.

- The regularization parameter α was also added to the grid-search as another dimension. This yielded the following hyperparameters for models which were tested. All models were tested both with the α value listed as well as with $\alpha = 0$.

Table 1: Hyperparameters of models

Input	Order	α



(a) (b)

Figure 3: Figure 3a shows an example of when the model performs well. Figure 3b shows an example of when the model breaks down.

Here the $input = 12$, $order = 2$, $\alpha = 0$ model performed best and the rest of the thesis will focus on this model.

- The performance of the model depends greatly on the data points it is trained on. An example of this can be seen in figure 3 where the $input = 12$, $order = 2$, and $\alpha = 0$ model is applied at different points along a set of trajectories. When looking at the histogram of residuals of the model shown in figure 4, we can see that they form a nice Gaussian distribution without any large outliers. We can therefore see that the model does not break down more than expected. The model has an MAE of ≈ 0.012. The MAE of the model increases exponentially which can be seen in figure 5 where the MAE of the model is also compared to an ML force field created by Chmiela et al. which has an MAE of ≈ 0.019. Here it can be seen that when the model is only predicting one step in the future it clearly has a lower MAE than Chmiela et al.'s force field. When the model predicts two steps in the future it performs slightly worse than the force field and after this point the exponential increase of error on the VAR model makes it perform far worse than the ML force field.

- When figuring out the speed-up achieved by the model the average time it takes to find the positions at the next time step was calculated for DFT steps, AR steps, and the model which combines DFT calculated and AR predicted steps. The average times can be seen in figure 6. The time it takes for the AR model to predict the forces and calculate the new positions and momenta is much smaller than the time it takes DFT. This means that we can approximate the speed-up gained from using the model by setting the time spent on the AR step to 0 and using the equation $\frac{input+pred}{input}$ which for the input $= 12$, lag $= 2$, equals $\frac{13}{12} = 1.0833$. The measured speed-up for that model is 1.0816.

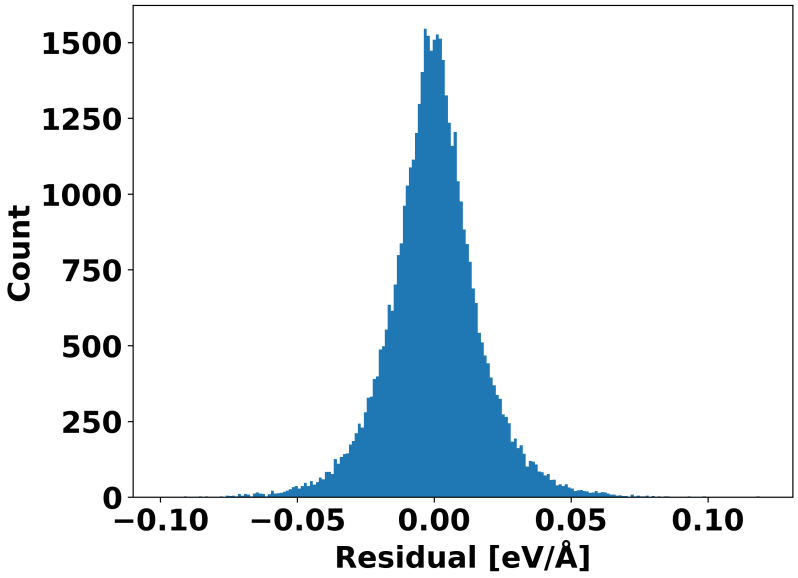


Figure 4: The figure shows the distribution of the input = 12, order = 2 model's residuals.

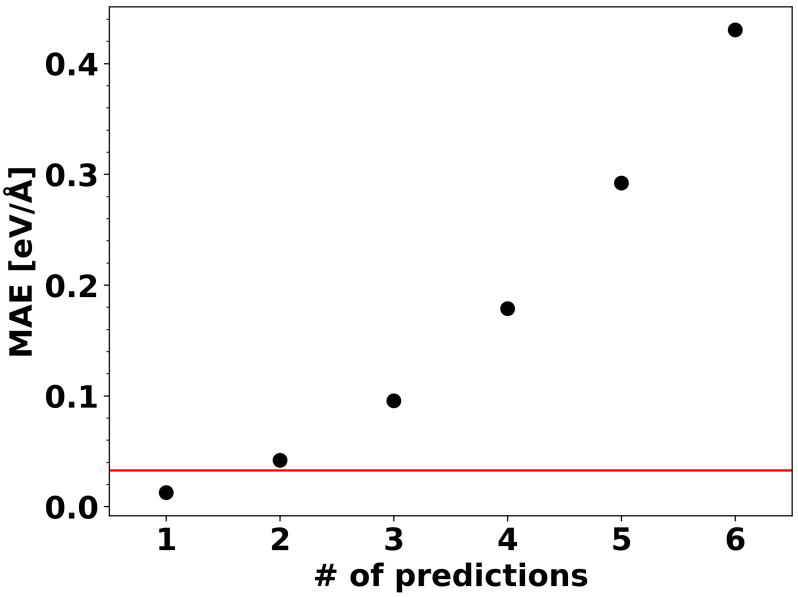


Figure 5: The figure shows the model's MAE for different numbers of predictions made. The red line corresponds to the MAE achieved by Chmiela et al.'s force field [2].

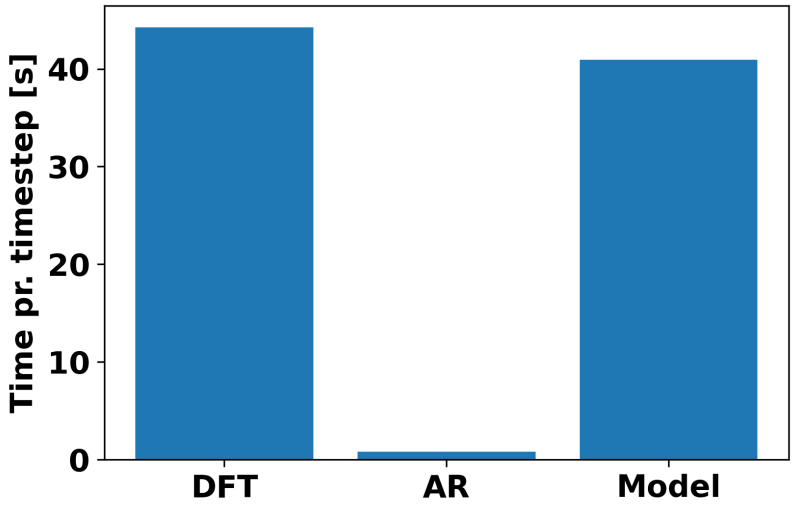


Figure 6: Bar plot showing the average times for a DFT step, model step, and a simulation incorporating the model.

- Ethanol in gas-phase was chosen as the system to investigate. All simulations were done with ASE and GPAW, using PBE as the exchange-correlation functional with single-zeta as the basis set. A time-step of $dt = 0.5\,\text{fs}$ and a temperature of $T = 400\,\text{K}$ was used to set the initial momenta of the atoms using a Maxwell-Boltzmann distribution. The Ethanol was optimized using BFGS and the simulation was done using the velocityVerlet algorithm by ASE which is an NVE ensemble.
  - Show math of velocityVerlet algorithm?

2.2 Metrics

- Since a molecule is a highly chaotic system it is difficult to compare simulations with different starting values due to how much they vary. While simple values, like average angles or bond lengths, might roughly average out over a simulation. However, more advanced metrics, like distributions, are unlikely to. Therefore, if you need to compare two simulations with each other they need to have the same random seed.

- We can therefore compare a pure simulation to a simulation using our model. However, any comparison is meaningless without any reference. There are two different ways we can create a reference for the comparison between simulation and model. The first and simplest way is to compare the simulation to another simulation with a different random seed. We can then see whether the variation is larger between simulations with different seeds or between a simulation and a model with the same seed. However, due to the large difference caused by the differing initial momenta, it should be easy for the variance between the model and simulation to be lower. This reference therefore gives a good minimum requirement but is not a thorough test of the model.

- A better reference is by increasing the time-step in the velVerlet algorithm to match the speed-up achieved by the model. The time step needed to match the speed-up is approximated by $dt_{new} = \frac{dt_{old} * (input + pred)}{input}$. If the model performs worse than simply increasing the speed-up of the dynamics algorithm increasing the timestep would yield better results than using the model.

- **MEAN VALUES.** The mean values of angles and bond lengths are the simplest way to evaluate the performance of a simulation. An advantage of the metric is that you can compare it to experimental values. However, the average values do not give us an idea of the molecule's geometry along the trajectory as the same average value can come from wildly different distributions.

- **DISTRIBUTIONS.** The distribution of angles and bond lengths would give a better understanding of the systems. However, because they only describe a small part of the molecule they are still not optimal in describing the geometry of a molecule. The geometry of the molecule depends on all of the angles in the molecule and choosing to look at a couple of angles or bond lengths leaves out a large part of the picture. A better way would be to look at dihedral angles which will be discussed later. Also, the distributions must mostly be evaluated by eye as mathematical methods, like the Kolmogorov-Smirnoff test, yield a small probability of the distributions matching for large datasets.

  - Q-Q/P-P plots?

- **RMSD.** Root mean square deviation (RMSD) is a way to measure how much a molecule differs at a timestep from a reference molecule. By looking at the RMSD you can evaluate the mean evolution of the molecule at each timestep. This way you can ensure that the overall temporal evolution of the molecule matches. One thing to keep in mind is the fact the RMSD is the average deviation of all atoms. It can therefore yield the same value for two molecules where different atoms have the same deviation. It can therefore not be used to say anything about the specific geometry of the molecule.

- **CONFORMATIONAL ISOMERS.** When you want to look at whether the geometry of the model trajectory keeps up the chemical properties one thing you could look at is conformational isomers. Ethanol has two different conformational isomers depending on where the hydrogen in the hydroxyl group points. The two isomers can be seen in figure 7. To evaluate which isomer the molecules in in at each time step the scalar projection between two vectors, C-C and O-H, is calculated.

- **2D DIHEDRAL** Creating a heatmap of the molecule's energy at two different dihedral angles is the most common way of evaluating the performance of an ML force field. Dihedral angles are better than angles to describe the molecule's geometry. By using the force field to calculate the molecule's energy at different configurations of dihedral angles, you make sure that the force field describes the correct energy landscape of the system. As our model does not calculate energy we can not do the exact same, but we have created a 2D histogram of the different dihedral angles to serve as a population map that can act as a proxy for the energy.

2.3 Results

- The mean values are quite similar between simulation, model, and velver speed up. Overall it seems that the velver speed up performs slightly better, however the model performs better at certain angles and bond lengths. Overall both the model and velver speed up is closer to a simulation with the same random seed than a simulation with a different random seed is.

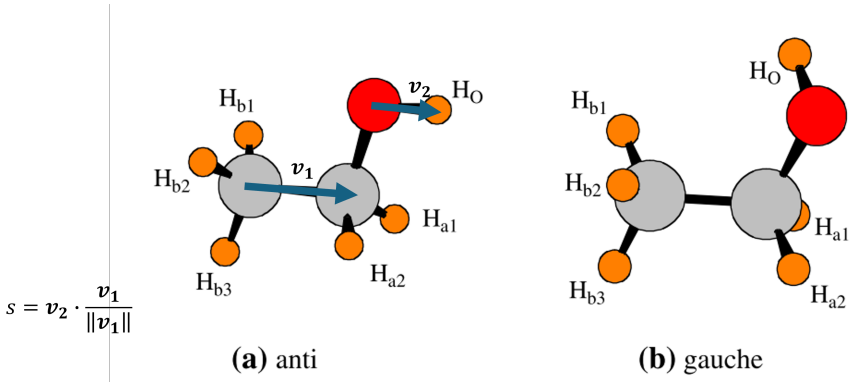
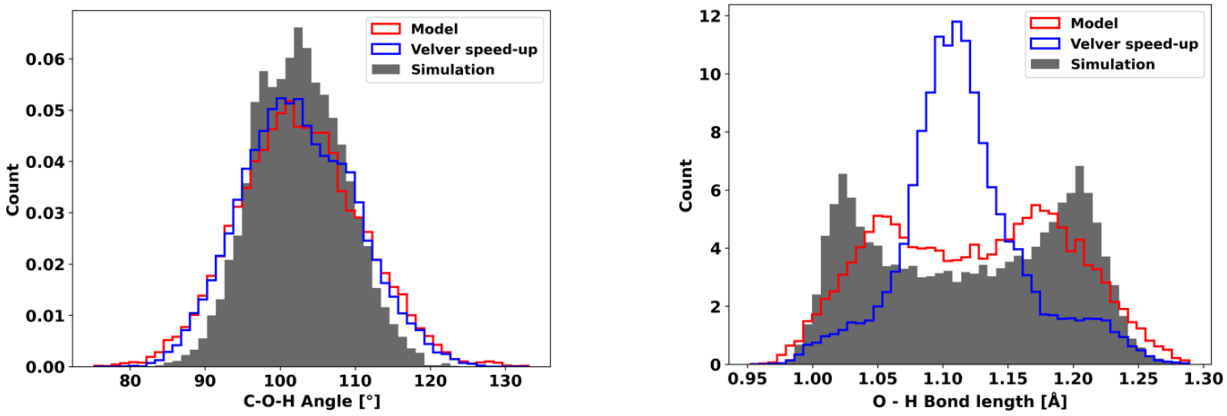


Fig. 1 *Anti* and *gauche* conformers of ethanol, showing atomic labeling used below

Figure 7: The two conformational isomers of ethanol are shown with the gauche isomer on the left and the anti-conformer on the right. An example of the calculation of vector projection used in identifying ethanol's isomer state is also shown. **NEED TO CREATE VERSION WITH NO COPYRIGHT ISSUES**



(a) (b)

Figure 8: Figure 8a shows the distribution of the C-O-H angle. Figure 8b shows the distribution of the O-H bond length. Both figures show data from a simulation, model, and velverlet speed-up with one seed, and a simulation with another seed.

- An example of a distribution of an angle and a bond length is shown in figure 8. Here is an example of an angle where both the distributions from the model and velver speed-up roughly match the distribution from the simulation, with both being very close to each other. When looking at the distribution for the O-H bond length we can see how average values can lie. All three trajectories had an average value that was close to each other but when we look at the distribution we can see that the model and simulation is split into two peaks, for the anti and gauche conformational isomers, while the velver speed-up only has a single peak in the middle. Figure 9 shows Q-Q plots of the same values.

- The RMSD is greatly dependent on the random seed as the initial momenta has a big influence on the drift of the molecule. It therefore only makes sense to look at trajectories with the same random seed. Of course this also means that the RMSD of the trajectories will start out being very similar but will devolve with time. Figure 10

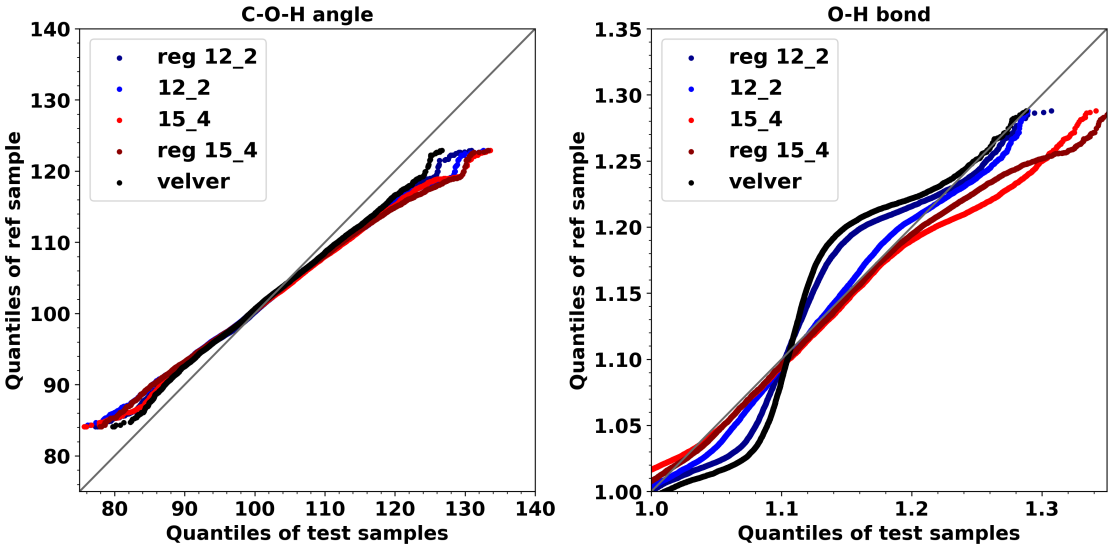


Figure 9: Q-Q plot of C-O-H angle and O-H bond length.

shows the RMSD of the trajectories after 500fs. Here it can be seen that the model has been decoupled from the simulation and only the general drift of the molecule remains, not any of its oscillations. On the other hand the velver speed-up trajectory is still very strongly coupled to the simulation.

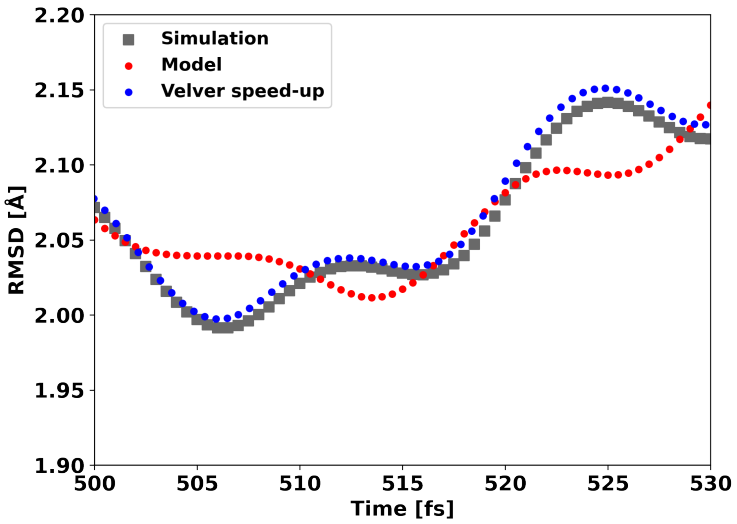


Figure 10: RMSD of trajectories created by simulation, model, and sped-up simulation. The RMSD is calculated between the molecule at each frame and the molecule in its original position in the first frame of the trajectory.

- Figure 11 shows the distribution of the scalar projection for the different trajectories. Again, both the model and the velver speed up come closer to the simulation with the same random seed than a simulation with a different random seed does. The model seems to perform slightly better than the velver speed up but there is not much separating the two.
- Figure 12 shows 2D histograms of dihedral angles of ethanol. The simulation shows that the population is contained in two clear lines for both seeds, the exact location

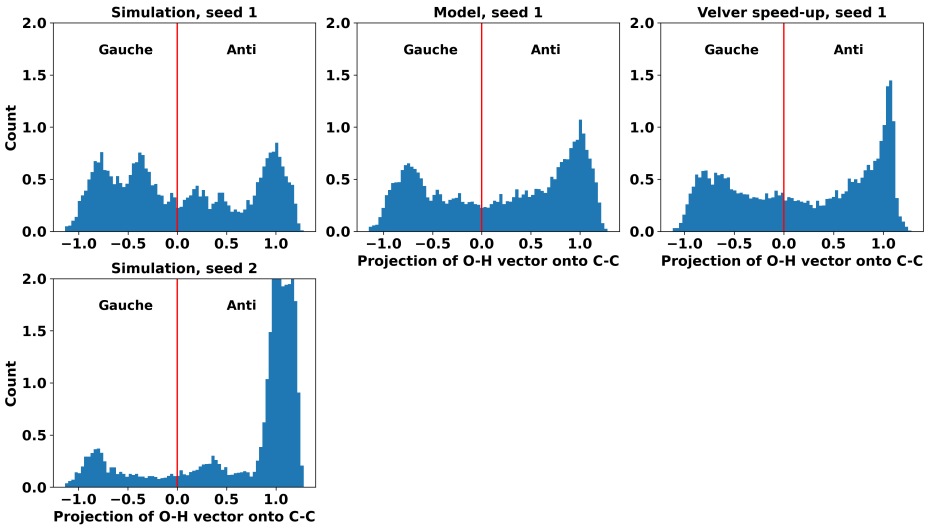


Figure 11: The distribution of the scalar vector projection is shown in a histogram for different trajectories. It is shown for the simulation, model, and sped-up simulation. All methods were used on two different seeds.

of the lines varies, probably due to the differing initial momenta, but both show two lines with a hotspot at a certain point on these lines. Compared to this the model's 2D histograms are much more smeared out and do not show clear lines. The Velver speed-up is more smeared out than the simulation, not showing the clear hotspots that are present in those, but it still clearly shows a horizontal and vertical line

2.4 Perspective

- An easy way to increase the speed-up of the model is to increase the number of predictions made. However, as was shown in figure 5 the MAE increases exponentially with the number of predictions made, and when compared to the force field by Chmiela et al. the model's MAE becomes much larger when predicting more than 3 steps. Models predicting up to 6 time steps were tested and can be seen in figure 13. Here it can be seen that the performance quickly deteriorates. When predicting two time steps the 2D histogram still kind of looks like when predicting 1 step, but when the model predicts any more steps than that it quickly becomes unrecognizable.

- In order to figure out whether the model was just inducing random error every time it predicted a step, simulations were run where DFT was used to calculate all forces but every 12 time steps a random Gaussian error with varying σ was added. The result can be seen in figure 14. The green line has a σ corresponding to the MAE of the model. It can be seen that the model performs much better than randomly induced error. **CHECK AT INDUCED ERROR SIM ER KORREKT.**

- We wanted to see if applying the model to larger molecules would yield better results. The two molecules tested were aspiring and cholesterol, as shown in figure 15. Cholesterol was chosen as a best-case scenario for the model. It has a lot of atoms, and most of them are carbon and hydrogen hopefully yielding a smooth energy surface that is easy for the model to predict. Aspirin was chosen as a heavy molecule with a more varying energy landscape. This was done so it could be tested how much it helped

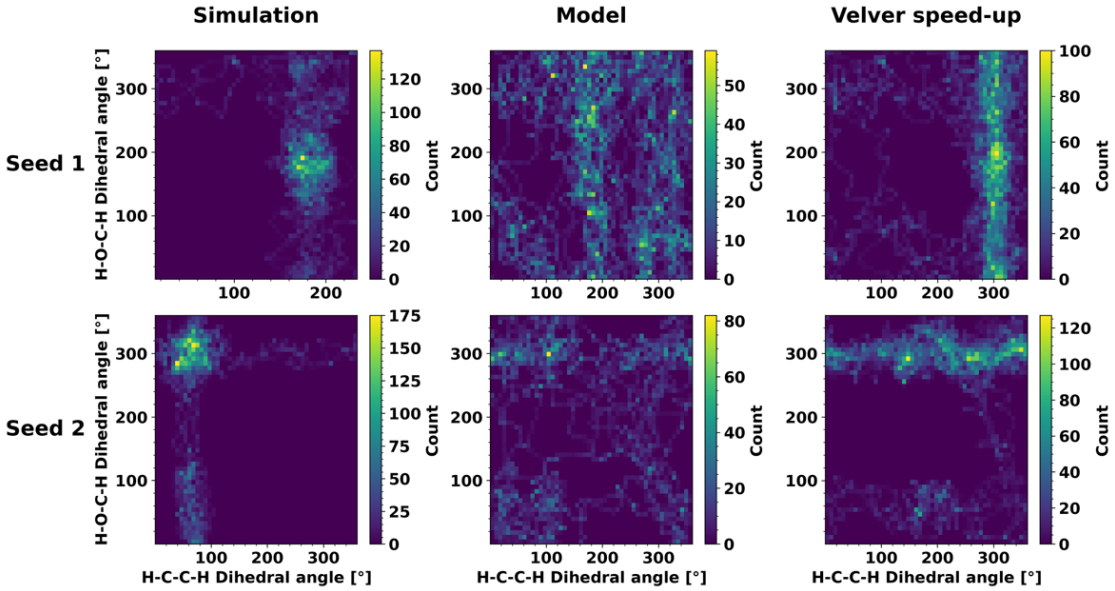


Figure 12: The figure shows the distribution of dihedral angles for various methods at two different start seeds

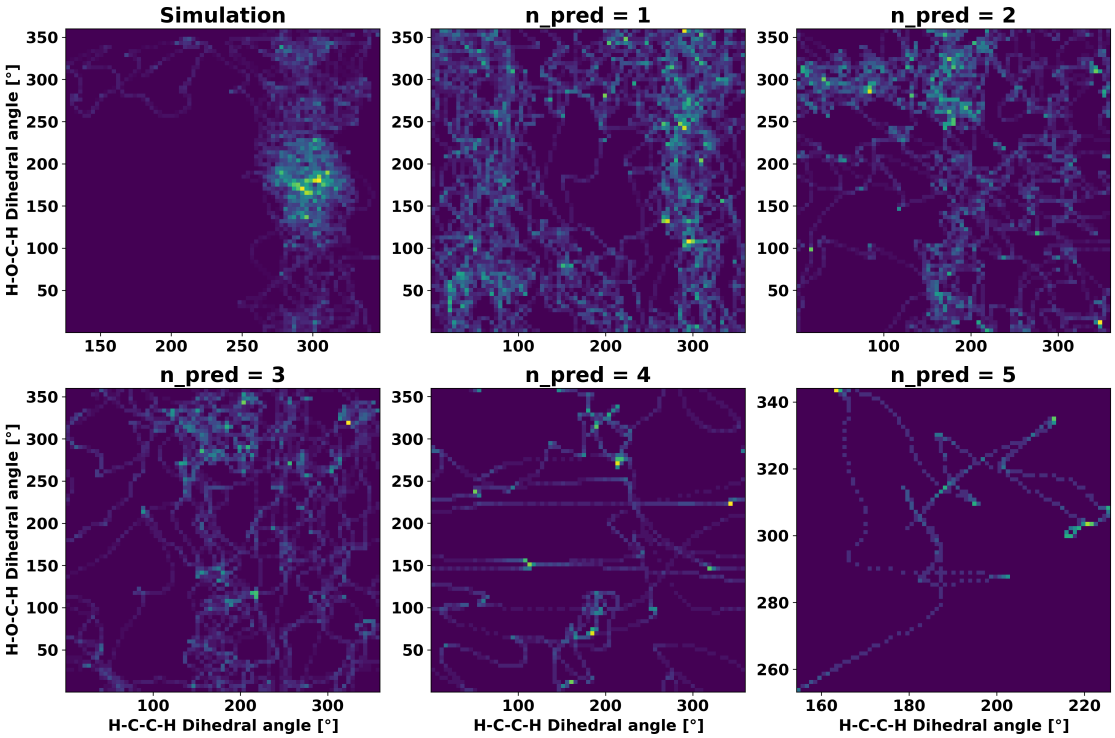


Figure 13: Dihedral angle 2D histograms for models with different number of predicted steps

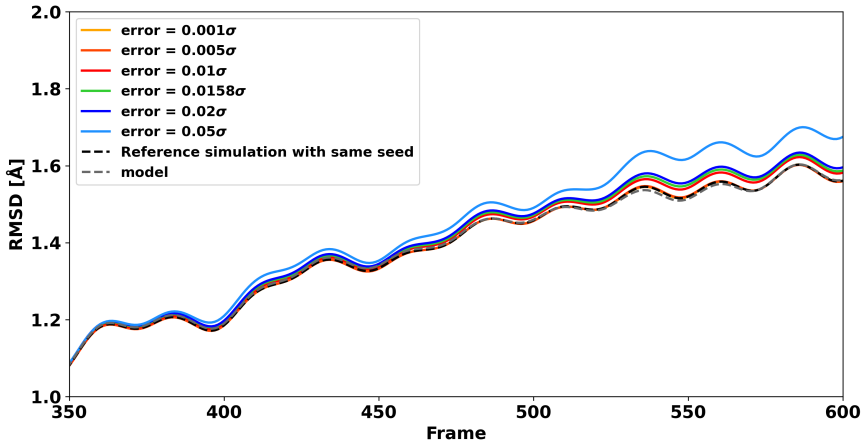
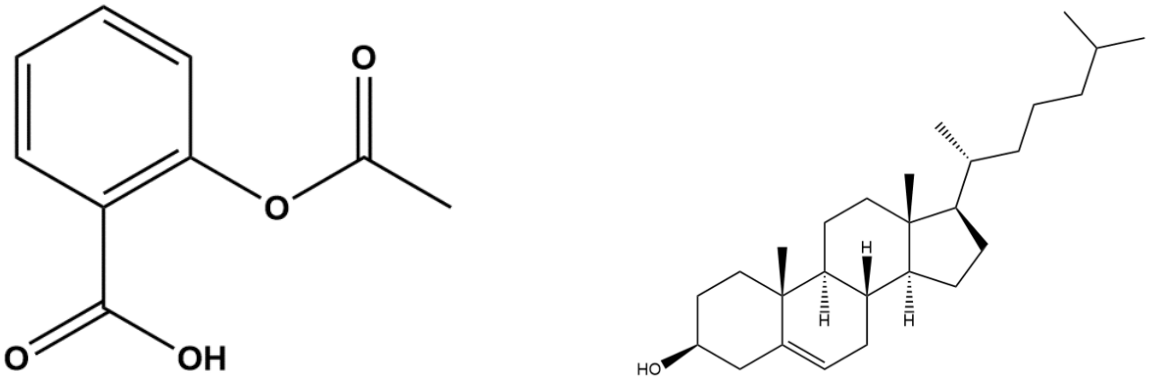


Figure 14: Comparing RMSD of model with simulation with induced error. **CHECK HOW DIHEDRAL 2D HISTOGRAM LOOKS.**



(a) (b)

Figure 15: Figure 15a shows aspirin while figure 15b shows cholesterol.

the model have access to a large amount of atoms that could act as a buffer to the model's error.

- The big difference between ethanol and heavier molecules appears when looking at the 2D histograms of dihedral angles. A figure showing the histograms of aspirin's dihedral angles can be seen in figure 16 while the histograms of cholesterol's dihedral angles can be seen in figure 17. The heavier molecules perform much better here than ethanol and there is no large smear for the heavy molecules. When comparing the performance of aspirin to cholesterol cholesterol performs slightly better than aspirin, but both perform better than ethanol. This suggests that the model performs better both on molecules with a larger amount of atoms, but also on molecules with a consistent energy surface. This is promising for the application of the model to materials where there is a large number of atoms generally in a periodic pattern yielding a consistent energy landscape across the material.

- Overall the model seems to perform worse on ethanol than just increasing the time step to achieve the same speed-up. There are some parts where the model performs better than velver speed-up most centered around the description of the hydroxyl groups as seen in the O-H bond length distribution and conformational isomer distribution.

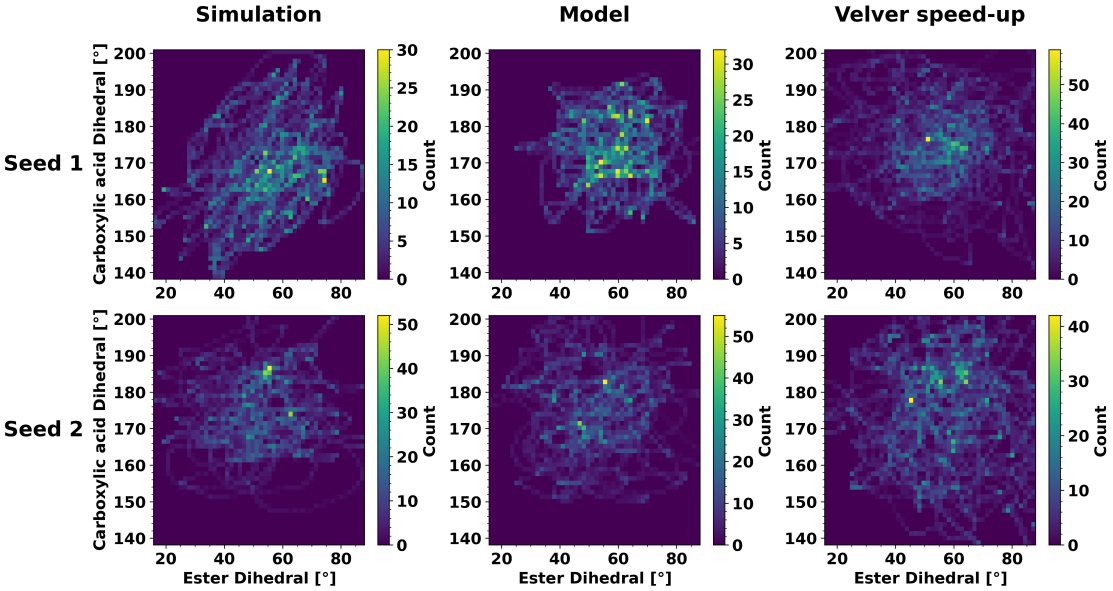


Figure 16: 2D Histograms of two dihedral angles of aspirin are shown for two different seeds and three different methods, simulation, model, and a simulation with larger timestep to achieve the same speed-up as the model. The two vertical lines are the different seeds while the horizontal lines are the different methods. **FIX COUNT**

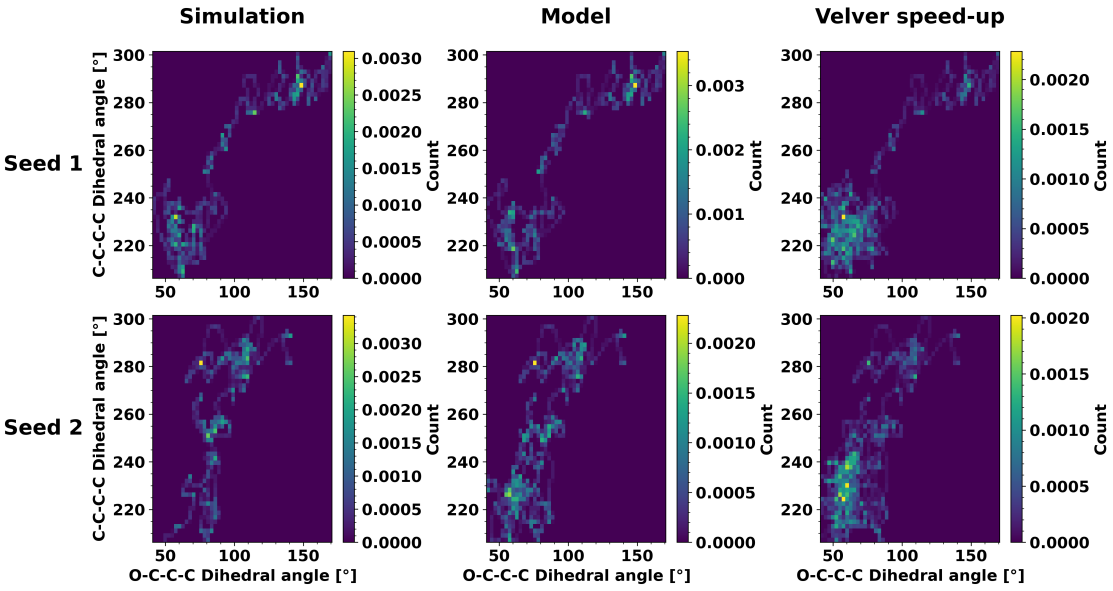


Figure 17: 2D Histograms of two dihedral angles of cholesterol are shown for two different seeds and three different methods, simulation, model, and a simulation with larger timestep to achieve the same speed-up as the model. The two vertical lines are the different seeds while the horizontal lines are the different methods. **FIX COUNT**

However as seen in the dihedral the overall geometry of the molecule seems much worse in the model compared to the velver speed-up. It would therefore be better to just increase the time step as you can get a comparable speed up with more accurate results and less work. The model does seem to actual use information from the force trajectories as seen in the fact that it outperforms induced error. Application of the model on heavy molecules also yield more promising results than its use on ethanol. This could be an avenue of further research and its application on materials could be promising due to their large amount of atoms and periodic energy landscape. Most likely the AR model is not advanced enough to properly utilize the information in the force trajectories, and a recurrent neural network (RNN) could possibly solve this while adding more information on the molecular environment from descriptors.

3 RNN

-

4 Perspective

-

References

[1] Jamshed Anwar and Dirk Zahn. Uncovering molecular processes in crystal nucleation and growth by using molecular simulation. *Angewandte Chemie International Edition*, 50(9):1996–2013, 2011.

[2] Stefan Chmiela, Alexandre Tkatchenko, Huziel E. Sauceda, Igor Poltavsky, Kristof T. Schütt, and Klaus-Robert Müller. Machine learning of accurate energy-conserving molecular force fields. *Science Advances*, 3(5):e1603015, 2017.