Generation of an accurate potential energy surface for carbonylsulfide in helium

Report on in-depth Practical
Richard J. M. Beckmann
Supervisor: Prof. Dominik Marx

August 26, 2019

1 Introduction

Helium, the lightest of noble gases, exhibits a number of unsusual properties. It is the only element besides its next-period analogue neon which does not show any chemical reactivity and remains liquid even at 0 K. Instead, it enters a superfluid state, Helium II. Characterized at first by its nearly ideal heat conductivity, it was suggested by Kapitza that this was merely an effect of rapid convection caused by the absence of internal friction [1]. This was experimentally shown in 1946 by Elephter Andronikashvili, who measured the moment of inertia of a rod immersed in cold helium. Upon decreasing the temperature the apparent moment of inertia increases until reaching the transition temperature T_λ, below which it would rapidly decrease again. It is assumed that this transition temperature T_λ marks the onset of superfluidity and that decreasing the temperature converts a growing fraction of the liquid into superfluid helium. Since superfluidity of ^{4}He hinges on bosonic exchange [2,3], it is strictly a cluster effect with no such thing as a single, superfluid helium atom, but emerging from the interaction of a high number of helium atoms. This gives rise to the question of the minimal cluster size necessary to produce this effect. Experimentally, Andronikashvili's experiment can be performed on small clusters by probing the rotational spectrum of single molecules like carbonyl sulfide (OCS) [4] with differing numbers of helium atoms solvating the molecule.

To compare these experiments to theoretical predictions and to analyze the physical effects in detail, however, accurate descriptions of the system are required. To use Molecular Dynamics (MD) or Monte Carlo (MC) methods, two of the most common tools to investigate molecular systems, the energy of a given configuration has to be made accessible. The most straightforward way to determine the energy of the molecule and the forces acting on each atom would be an *ab-initio* MD simulation in which a single point electron strucutre calculation is performed after each step. This approach yields highly accurate energies and forces, but it comes at a high computational cost. Paying this high cost is made even more painful by the fact that many configurations are visited more than once, each time calculating repeating the same calculations. A recent solution for this dilemma has arisen in form of neural network potentials (NNPs). These use a machine learning approach to fit a neural network (NN) to a set of known configurations, from which the energy of other configurations can be interpolated. This neural network can then be used to run an MD simulation at very low computational cost while retaining the high accuracy of the method used in its generation.

In this work, the process of generating, refining and validating a neural network describing OCS solvated in helium will be described. Section 2 will cover the structure and characteristics of a neural network trained for the prediction of atomic energies. Specifically the use of symmetry functions as input vectors will be discussed. Then, in section 3, the iterative enhancement of the training set over several generations is described. The NNP acquired in this way was extensively validated as described in section 4. Computational details on these processes is given in section 5. The manuscript concludes with a summary of the work and an outlook on further investigations in section 6.
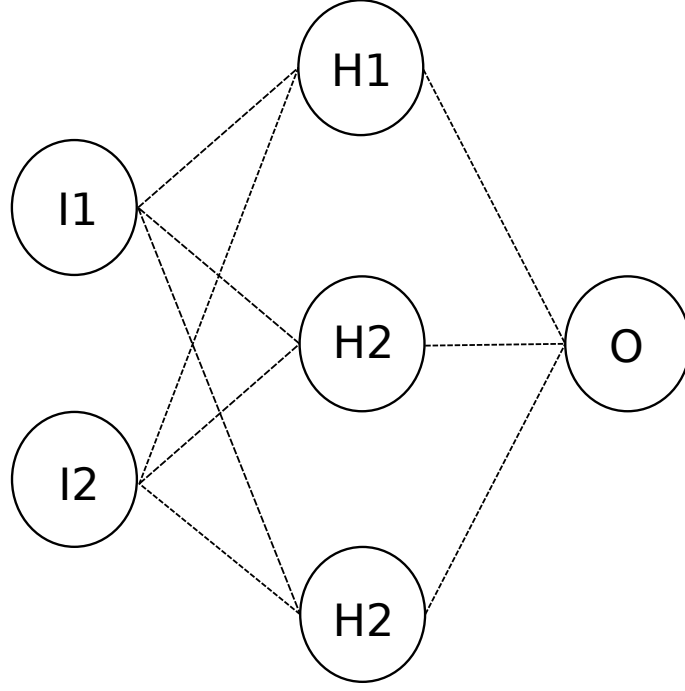
Figure 1: Example of a simple neural network. I1 and I2 are the input nodes, H1-H3 the hidden nodes and O the output node. Dashed lines show the connections between nodes.

2 High Dimensional Atomic Neural Network Potentials

[5–8] An artificial neural network consists three type of layers: The input layer, the output layer and a variable number of hidden layers. The nodes in each layer, the "articifial neurons", are connected with each node in the previous layer and each node in the subsequent layer. The input vector $\underline{I}_l$ received by layer l is always a product of the previous layers' $l-1$ output vector $\underline{O}_{l-1}$ and the weight matrix connecting these layers $\underline{\underline{W}}_l$, modified by an offset vector $\underline{b}_l$

$$\underline{I}_l = \underline{b}_l + \underline{\underline{W}}_l \cdot \underline{O}_{l-1} \tag{1}$$

The exception from this is the input layer, which receives it's values as an input vector that the NN is meant to process. Additionally, each node may apply some activation function $f_{l,m}$ to the value received to generate it's output value

$$\underline{O}_l = \underline{f}_l(\underline{I}_l) \tag{2}$$

A schematic depiction of a neural network is shown in figure 1.

This basic structure of a feed forward network is adapted for the creation of a high dimensional atomic neural network potential. The total energy of the system in such a network is calculated as a sum of atomic energies which are gained by applying an element specific neural network to each atom. This is schematically shown for a protonated cluster of two water molecules in figure 2
Choosing the form of the input vector is one of the most critical choices when creating a NN. In principle, the cartesian coordinates of the system under consideration
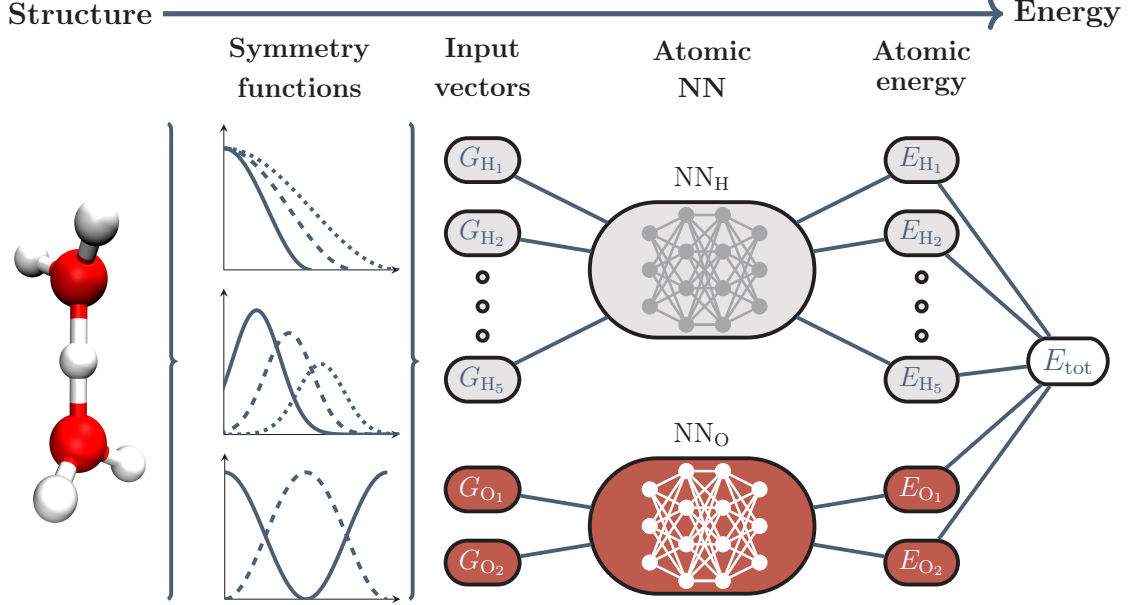
Figure 2: Application of a high dimensional atomic neural network potential to a water cluster. The configuration of the system is converted into a description via symmetry functions. These input vectors - one for each atoms in the system - are the fed into the NN specific for the respective atom's element. Thereby an energy output is generated for each atom, the sum of which is the toal energy of the system.

could be used as input vector from which the neural networks predicts the energy of the system. This, however, comes with several severe drawbacks. For once, the energy of a system is invariant with respect to translation and rotation of the system. This means that, when using cartesian coordinates as input, an ininite number of different inputs will yield the same total energy. Samewise, the energy must not remain constant if two atoms of the same element are exchanged.

To enforce these symmetries, the system with interatomic distances and angles between atoms could be described in intramolecular coordinates. However, if larger systems are to be observed, this would make the use of cut-off radii impossible. since the number of inut functions would vary with the number of atoms within the cut-off radius. Instead, it has repeatedly been shown to be prudent to use a set of radial and angular symmetry functions for each atom, the value of which depends on the number and position of atoms within the immediate vicinity of the atom under consideration. The cut-off function is given by:

$$f_C = \begin{cases} 0.5 \cdot [cos(\frac{\pi R_{ij}}{R_C}) + 1] \text{ for } R_{ij} < R_C \\ 0 \text{ for } R_{ij} > R_C \end{cases} \quad (3)$$

where R_C is the cut-off radius and R_{ij} the distance between atoms i and j. In this work, a cut-off radius of $R_C = 12 \text{ Å}$ was chosen. To describe distances between atoms, radial symmetry functions of the following form were used:

$$G_i^R = \sum_{j \neq i}^{N} e^{-\eta R_{ij}^2} \cdot f_C(R_{ij}) \quad (4)$$

where η is a parameter manually chosen to modulate the selectivity of the function for atoms at different radii.

The second type of symmetry functions used is the angular symmetry function of the form

$$G_i^\theta = 2^{1-\zeta} \sum_{j\neq i}^{N} \sum_{k\neq i,j}^{N} (1 + \lambda \cos\theta_{ijk})^\zeta e^{\eta(R_{ij}+R_{ik}+R_{jk})^2} \cdot f_C(R_{ij}) \cdot f_C(R_{ik}) \cdot f_C(R_{jk}) \quad (5)$$

An overview over all symmetry functions used can be found in table XXX.

3   Fitting Procedure

To generate a PES for OCS solvated in helium, the potential is split into three parts:

1. The intramolecular interaction between oxygen, carbon and sulfur

2. The intermolecular interaction between helium and OCS

3. The intermolecular interaction between helium atoms

This splitting is expedient for several reasons: For once, only the first two of these potentials are described by NNs, whereas the helium-helium interaction can be given by an analytical form [9]. This reduces computational cost by a large margin for systems with more than a few helium atoms. Furthermore, this allows the use of the network for different numbers of helium atoms. If the whole system was described with one NN, a new network would have to be fitted everytime the number of atoms is changed. By this separation of potentials, however, more atoms can be included in the simulation and their interaction energy simply added to the total energy of the system, neglecting only 3-body- and higher interaction terms.

To find a NNP describing the OCS molecule without helium, at first a DFT *ab-initio* trajectory at 300 K was run. From this trajectory, configurations of the molecule were chosen at random and CCSD(T)-F12a calculations with the AVTZ basis set were performed. Two NNs were trained and tested on these points. As with all NNs created in this work, these NNs possessed two hidden layers with 25 nodes in each layer. To adjust the weights $w_{l,m,l+1,k}$, backpropagation of errors [10] was used. With only 40 points, phase space is poorly sampled, even for a relatively simple molecule like OCS. Due to the highly flexible nature of NNs, the two networks will disagree strongly on the energy of configurations which lie in these undersampled areas. This can be utilised to identify these areas and systematically add points to the training set. To do so, a large number of configurations were chosen from the trajectory and the energy predicted by both NNs. For the 40 points for which the difference between predictions were strongest, new CCSD(T) calculations were performed and the points added to the training set. By repeating this procedure, several generations with an iteratively increasing training set were built. (MAYBE I SHOULD SHOW THE DEVELOPMENT OF THE MEAN SQUARE ERROR AGAINST THE GENERATION?) Since this training set was still based on only the 300 K simulation, the NN was used to perform additional simulations with different simulation methods at temperatures between 1 K and 1000 K to sample phase space. The parameters used in these simulations can be found in section 5. To combine these NNs into one final network, the final sets of training points from each condition were accumulated and a NN trained on those points.

Table 1: Comparison between the vibrational frequencies of the molecule in the minimum energy configuration according to the NNP and CCSD(T)-F12a reference calculations. The RMSD is 6.89 cm^{-1}

NNP	CCSD(T)-F12a	difference
529.50	519.76	9.74
529.50	519.94	9.56
874.10	873.87	0.23
2086.15	2084.67	1.84

The final set of training points acquired under each of those conditions was then used to generate a NN describing the OCS helium interaction. A small number of OCS configurations was taken from this set of points and one helium atom was added to the system at a random position. As before, the energy of these configurations was determined via CCSD(T)-F12a and two interaction NNPs were fitted to these points. In subsequent generations, the training set was iteratively and systematically increased. This was achieved in a similar fashion as before by generating a large number of testing points from which the new points were taken. However, the candidate points were not generated via Molecular Dynamics but Monte Carlo simulations, starting from one of the already existing training points. After ccepting the specified number of moves, the NNs are used to predict the interaction energy at those points. A number of points with strongest disagreement are chosen for CCSD(T)-F12a calculations and added to the training set. After performing this procedure for each of the different conditions, the training sets were combined and a final network fitted to this combined set.

4   Validation

To test the PES generated for OCS, a normal mode analysis was performed. First, the molecule was geometry optimized using a conjugate gradient algorithm using CCSD(T)-F12a calculations and the NNP. The diagonalized Hessian matrix yielded the normal mode frequencies shown in table **??**. The root mean square deviation (RMSD) between the NNP based calculation and the reference calculation is 6.89 cm^{-1}.

5   Computational Details

I would like to put here the computational details so they do not clog up the text in "Fitting the Neural Network", but maybe it would be more straightforward/natural to put them there? What do you think?

6 Summary and Outlook

References

[1] P. KAPITZA. Viscosity of Liquid Helium below the λ-Point. *Nature*, 141(3558):74–74, jan 1938.

[2] F. LONDON. The λ-Phenomenon of Liquid Helium and the Bose Einstein Degeneracy. *Nature*, 141(3571):643–644, apr 1938.

[3] Adrian F. G. Wyatt. Evidence for a Bose Einstein condensate in liquid 4He from quantum evaporation. *Nature*, 391(6662):56–59, jan 1998.

[4] S. Grebenev. Superfluidity Within a Small Helium-4 Cluster: The Microscopic Andronikashvili Experiment. *Science*, 279(5359):2083–2086, mar 1998.

[5] Jörg Behler and Michele Parrinello. Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces. *Physical Review Letters*, 98(14):146401, apr 2007.

[6] Jörg Behler. Neural network potential-energy surfaces in chemistry: a tool for large-scale simulations. *Physical Chemistry Chemical Physics*, 13(40):17930, 2011.

[7] Jörg Behler. Constructing high-dimensional neural network potentials: A tutorial review. *International Journal of Quantum Chemistry*, 115(16):1032–1050, aug 2015.

[8] Jörg Behler. First Principles Neural Network Potentials for Reactive Simulations of Large Molecular and Condensed Systems. *Angewandte Chemie International Edition*, 56(42):12828–12840, oct 2017.

[9] Ronald A. Aziz, Alec R. Janzen, and Michael R. Moldover. Ab Initio Calculations for Helium: A Standard for Transport Property Measurements. *Physical Review Letters*, 74(9):1586–1589, feb 1995.

[10] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, (323):533–536, 1986.