Supplementary Information of “Transfer-Learning-Based Coarse-Graining Method for Simple Fluids: Toward Deep Inverse Liquid-State Theory”

Alireza Moradzadeh and Narayana R. Aluru

aluru@illinois.edu

Department of Mechanical Science and Engineering, Beckman Institute for Advanced Science and Technology, University of Illinois at Urbana–Champaign, Urbana, Illinois, 61801 United States
S.1. Thermodynamic Properties

Given that the deep neural network (DNN) obtains the solution to the inverse-problem of liquid-state theory, we investigate the estimation of thermodynamic properties of Lennard-Jones particles using the DNN and compare them with values obtained from MD simulations of Argon case as discussed in the main manuscript (Argon at various temperatures and densities, see Figure S1 for RDFs of Argon at various thermodynamic states). For example, pressure of a given molecular system can be calculated using,

\[ P = \frac{1}{V} \left( \frac{1}{3} k_B T + \frac{1}{3} \sum_{i>j} f_{ij} \cdot r_{ij} \right) \]  

(S1)

where \( N_{\text{DOF}} \) is the number of degrees of freedom in the system, and \( V \) is the volume of the simulation box. The second term in Eq. S1 is the virial part of the pressure and \( f_{ij} \) is the force between two beads, \( i \) and \( j \), which can be expressed in terms of the RDF and the pair potential. The pressure can then be expressed as,

\[ P = \rho k_B T - \frac{4\pi \rho^2}{6k_B T} \int_0^{r_{\text{cut}}} \frac{\partial u}{\partial r} g(r) r^3 dr \]  

(S2)

In a similar manner, one can calculate the total potential energy of a system based on the radial distribution function and the pair potential, as expressed below,

\[ U = \frac{N}{2} 4\pi \rho \int_0^{\infty} u(r) g(r) r^2 dr \]  

(S3)

where \( N \) is the number of particles in the system. Note that during the calculation, we consider dispersion correction for the virial and energy (see Reference 1\(^1\)) in a similar manner to that implemented in GROMACS.\(^2\) The comparison between pressure and energy predicted using MD and DNN is shown in Figure S2 and S3, respectively. The mean relative error for deep learning is less than 14% and 25% for the potential energy and pressure, respectively, with an average relative error of 13% and 17%, indicating the ability of the deep learning approach to predict the underlying physics and obtaining the solution to the inverse-problem of liquid-state theory for Lennard-Jones particles.
Figure S1. Radial distribution function of Argon at 121 different thermodynamic states obtained by uniformly sampling both temperature and density range shown in Table 1 of the main manuscript.

Figure S2. Bulk pressure of Argon for different thermodynamic states. a. MD results b. deep learning results.
Figure S3. Total potential energy of Argon for different thermodynamic states. a. MD results b. deep learning results
S.2. Additional Details on the Deep Learning Framework

The mean-squared error is used as the loss function to train the DNN. Optimization is performed using the Adam optimizer. The dropout technique is used for the second hidden layer, which resembles the regularization algorithm in reducing the overfitting significantly (see reference 3 for more details on the dropout technique). The dropout technique randomly drops nodes in layers with a specified probability, thereby reducing the number of active nodes in that layer. Dropout technique prevents nodes from undesirable co-adapting. The key idea in dropout is that in each minimization step of DNN instead of training the whole network only a fraction of the active nodes is used in training (see reference 4 for more information regarding deep learning).

We have tested multiple networks with different activation functions and size to select the DNN architectures according to Figure 1c of the main manuscript. A total of 36 networks are trained to choose the best architecture with the smallest loss function and least overfitting. All the networks used similar input (material fingerprint) except for the expansion power in temperature and density. Figure S4 shows the DL network with the best performance with two hidden layers and 48x15 size and a power of 3 ($p = 3$).

We also trained networks with 3 hidden layers, but these networks had an overfitting problem, as their performance over training and validation datasets diverged, i.e., it has a similar loss function value as the 2 layers network on the training dataset, but the loss function value is higher for the validation and testing datasets, indicating overfitting.
Figure S4. The loss function value of DL network for both training (black line) and validation (red dash-dot line) datasets. The value of loss function for testing dataset (blue dash-dot line). a. initial steps of training shown in log-scale b. final steps of training.

The output of FNN with three hidden layers uses the following sequence of equations,

\[
\begin{align*}
\mu_1 &= \tanh(W_1 x + b_1) + \text{sigmoid}(W_1 x + b_1) \\
\mu_2 &= \tanh(W_2 \mu_1 + b_2) + \text{sigmoid}(W_2 \mu_1 + b_2) \\
\mu_3 &= \tanh(W_3 \mu_2 + b_3) + \text{sigmoid}(W_3 \mu_2 + b_3) \\
v_o &= \text{sigmoid}(W_o \mu_3 + b_o)
\end{align*}
\]  

The accuracy of DNN is determined using the equation,

\[
a_{\text{DNN}, j} = \frac{\sum_{i \in D} f_{ij}}{|D|}
\]  

where \(a_{\text{DNN}, j}\) is the accuracy of the DNN for the potential parameter \(j (C_6, C_{12})\). \(|D|\) is the size of the dataset.

\(f_{ij}\) is determined based on how far is the prediction from the ground truth compared to square root of the loss function value on the training dataset, i.e.,

\[
f_{ij} = \begin{cases} 
1 & \text{if } |v_{j,\text{DNN}}^{(i)} - v_{j,\text{GT}}^{(i)}| \leq 4\epsilon L^{0.5} \\
0 & \text{otherwise}
\end{cases}
\]
where $\epsilon_L$ is the loss function value on the training dataset, which shows the interval for which 99 percent of DNN predictions are located from one-to-one mapping line, \( i.e., f_{ij} \) counts the number of data points within two parallel lines with the one-to-one mapping line, both of which have an offset of $\pm 4.0\epsilon_L^{0.5}$.

The mean absolute percentage error\(^5\) (MAPE) used in the main manuscript can be expressed as,

$$
\epsilon_{MAPE,j} = 100 \times \frac{\sum_{i \in D} |v_{j,DNN}^{(i)} - v_{j,GT}^{(i)}|}{\sum_{i \in D} |v_{j,GT}^{(i)}|} 
$$

(5.10)

The MSEs during training for both 2 layer and 3 layer networks are shown in Figure S5 and S6.
Figure S5. Loss function during training of 2 hidden layer networks. First column shows the $p$ (power of density and temperature for input) and first row shows the number of nodes in the first and second layers, respectively.
Size (1st, 2nd, and 3rd layers)

- 48x24x12
- 60x30x15
- 80x40x20

$p$

- 1
- 2
- 3
Based on the above results, we have selected three 2-hidden-layer networks (60x15 with power of 1 and 6 and 48x15 with powers of 3). The performance of all these networks is similar, but the second network performs better for the transferability case. Therefore, we used it for all the results shown in the main manuscript. Figure S7 and S8 show final values of MSE on each dataset for different network sizes and different features, which, in turn, justifies the selection of the aforementioned networks.
Figure S7. Final loss function values for training, validation, and testing datasets for various exponents ($p$) of temperature and pressure.
Figure S8. Final loss function values for training, validation, and testing datasets for various exponents ($p$) of temperature and pressure.
The transferability dataset includes data outside the range mentioned in Table 1 of the main manuscript. The data shown for the one-to-one mapping of the transferability case are obtained using the following steps. Based on the loss function value of the training dataset, data which are bounded by the accuracy line are selected. Then, minimum and maximum of the non-dimensional temperature \( T^* = \frac{k_B T}{C_{6,p}/4C_{12,p}} \), \( k_B \) is the Boltzmann constant and subscript \( p \) indicates predicted value by DL network) and density \( \rho^* = \frac{N(C_{12,p}/C_{6,p})^{1/3}}{L} \), \( L \) and \( N \) are the simulation box length and number of atoms, respectively) are determined for the selected data within two accuracy lines in order to obtain criteria to create a validity range for the non-dimensional temperature and density. Data points in the transferability dataset, which are within the validity range, are shown for each network including the results shown in Figure 2d and 2h of the main manuscript with non-dimensional temperature and density of \([0.42, 39.20]\) and \([0.0781, 0.802]\), respectively. The error distributions of all the three networks are shown in Figure S9.
Further assessment for the best DNN is performed for randomly selected data points (both potential parameters and thermodynamics states) of LJ atomistic particles, shown in Figure S10. We observe that DNN predicted potential parameters lead to almost indistinguishable RDFs compared to corresponding reference data.
Figure S10. Comparison between RDFs of DNN-based (solid red line) parameterized LJ potential and reference MD (dash-dot black line). Points are selected randomly from the dataset.
Comparison between DNN based coarse-graining and two other coarse-graining methods, namely, relative entropy and simplex, are shown in Figure S11 as well as in Figure 4 of the main manuscript. Furthermore, we compare the relative pressure (with respect to the AA reference) from the potentials parameterized using various methods, including the additional united-atom (UA)⁶,⁷ force field for methane (CH₄) (see Table S1). Figure S12 shows comparison of RDF between UA, AA and DNN models. Even though the pressure obtained using the UA method is closer to AA model compared to the DNN method (the pressure of UA and DNN models are 0.84 and 0.72 of AA pressure), the RDF of UA model differs in magnitude and location of peak from the AA model. The RDF of DNN model matches the magnitude and location of the peaks of the AA model.

**Figure S11.** Comparison between DNN-based (solid red line), relative entropy (dotted green line) and simplex (dash-dot blue line) coarse-graining models. All the three methods show excellent match with the all-atom model results (shown as black circles).

<table>
<thead>
<tr>
<th>Relative Pressure</th>
<th>( P_{\text{DeepILST}}/P_{\text{AA}} )</th>
<th>( P_{\text{Simplex}}/P_{\text{AA}} )</th>
<th>( P_{\text{RE}}/P_{\text{AA}} )</th>
<th>( P_{\text{UA}}/P_{\text{AA}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO</td>
<td>1.46</td>
<td>1.23</td>
<td>1.29</td>
<td></td>
</tr>
<tr>
<td>F₂</td>
<td><strong>1.30</strong></td>
<td>1.49</td>
<td>1.60</td>
<td></td>
</tr>
<tr>
<td>CH₄</td>
<td>0.72</td>
<td>0.66</td>
<td>0.80</td>
<td><strong>0.84</strong></td>
</tr>
</tbody>
</table>

Table S1. Comparison of the relative pressure obtained with various methods.
Figure S12. Comparison of RDF obtained with DNN (solid red line) and UA (dash-dot blue line) models with the all-atom methane (CH4) model (shown with black circles).
References


