Molecular Dynamics Properties without the Full Trajectory: A Denoising Autoencoder Network for Properties of Simple Liquids

Alireza Moradzadeh and N. R. Aluru*

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

ABSTRACT: Molecular dynamics (MD) simulation is a popularly used computational tool to compute microscopic and macroscopic properties of a variety of systems including liquids, solids, biological systems, etc. To determine properties of atomic systems to a good level of accuracy with minimal noise or fluctuation, MD simulations are performed over a long time ranging from a few nanoseconds to several tens to hundreds of nanoseconds depending on the system and the properties of interest. In this study, by considering simple liquids, we explore the feasibility of significantly reducing the MD simulation time to compute various properties of monatomic systems such as the structure, pressure, and isothermal compressibility. To do so, extensive MD simulations are performed on 12,000 distinct Lennard-Jones systems at various thermodynamic states. Then, a deep denoising autoencoder network is trained to take the radial distribution function (RDF) from a single snapshot of a Lennard-Jones liquid to compute the mean, temporally averaged RDF. We show that the method is successful in the prediction of RDF and other properties such as the pressure and isothermal compressibility that can be computed based on the RDF not only for Lennard-Jones liquids at various thermodynamic states but also for various simple liquids described by exponential, Yukawa, and inverse-power-law pair potentials.

Molecular dynamics (MD) is a popularly used computational tool to compute microscopic and macroscopic properties and understand physics of atomic systems in many areas of science and engineering including soft and hard matter.1 The Newtonian equations of motion are solved numerically by MD simulation to determine the trajectory of every atom in the system as a function of position and time. In the case of liquids, which is the problem considered in this work, the atomic trajectories are used to compute properties such as the radial distribution function (RDF), pressure, isothermal compressibility, etc. Among these properties, RDF is an important quantity as it not only characterizes the local structure of the atomistic system but can also be used to calculate thermodynamic properties such as the pressure, energy, and isothermal compressibility.5 To estimate these properties, MD simulations are performed over very long times (ranging from several nanoseconds to several tens to hundreds of nanoseconds depending on the property and system of interest). Once equilibrium of the atomic system is attained, physical properties are computed using data from each MD snapshot and averaged over all the MD snapshots to minimize fluctuation in the estimated property. Several attempts have been made to incorporate signal processing and other statistical concepts with MD simulation to reduce fluctuation and noise in estimated properties.3−8 However, signal processing and Fourier series expansion-based methods generally suffer from mathematical complexity and poor performance in the context of atomistic scale simulation. Reducing the computational time to calculate various properties of atomistic systems can not only accelerate fundamental studies on soft- and hard-matter but also accelerate the development of bottom-up coarse-grained models7−12 where an accurate calculation of structural and thermodynamic properties is required and is commonly achieved through long and repetitive MD simulations.

MD simulations generate a tremendous amount of data, which has not been exploited to a great extent until recently. The advent of concepts such as physics of big data and data-driven methods13 during the last five years has opened up opportunities to address the problem of obtaining accurate RDF (and other properties, e.g., thermodynamic properties) from a short MD simulation instead of a long one. Furthermore, methods such as hybrid Monte Carlo14,15 for systems with slow relaxation can generate extensive data sets with rich physics such as the liquid−liquid phase transition16,17 and nucleation.18 Among the various data-driven methods, deep learning is a promising method, which has been successfully employed in various fields, including atomic scale simulation.13,19−26 Various deep neural networks have
been developed, among which, autoencoder networks learn to encode data into a low-dimension representation ignoring noise in the data. Autoencoder networks map the data into a low dimension, and then they map the low-dimension encoding back to the dimension of the data in an unsupervised manner. Recent studies have successfully employed various types of autoencoder networks in atomistic scale simulation. Among the various autoencoder networks, the deep denoising autoencoder (DAE) is an autoencoder network in which the input data is intentionally corrupted by adding additional noise, which, in turn, improves the robustness of the DAE network training compared with simple autoencoder network.

In this study, we explore the possibility of using the DAE network to estimate RDF of simple liquids using a single snapshot from MD instead of hundreds of snapshots. RDF obtained from a single snapshot of MD simulation has large fluctuations, which mimics the noise used in the DAE training. In particular, the DAE used in this study learns to map a single snapshot RDF into a temporally averaged RDF (obtained through averaging of RDF over a long MD simulation). Note that the fluctuations in a single snapshot RDF are inherent features of MD data. To train the DAE network, large-scale MD simulations of 12 000 Lennard-Jones pair potentials at various thermodynamic states are carried out with a total simulation time of 24 µs. The data is generated through calculation of 800 single snapshots and corresponding temporally averaged RDFs of 12 000 Lennard-Jones systems summing up to 9.6 million single snapshot RDFs. Once the data is generated, it is used to train the DAE network through minimization of a loss function between DAE predicted and corresponding temporally averaged RDFs; i.e., the DAE learns to map any single snapshot RDF to its corresponding temporally averaged RDF. Furthermore, we combine traditional dimension reduction methods, specifically principal component analysis, with the DAE network. The objective of this approach is to take advantage of simplicity of a traditional method to reduce the size and training time of the DAE network and to increase accuracy of the traditional method to the level of the deep learning-based method. In this scheme, we change the DAE input vector from RDF to values of principle components.

Once the DAE network is trained, we assess its performance in prediction of RDF of Lennard-Jones systems based on a single snapshot RDF of MD simulation. We investigate the generalizability of the DAE network to other pair potentials describing monatomic systems belonging to the class of simple liquids. In particular, we investigate the exponential, inverse-power-law (IPL), and Yukawa pair potentials. The methodology can be extended to other phases of matter as well as to complex systems described by, for example, the Stillinger–Weber potential with interesting physics such as liquid–liquid phase transition.

The rest of the paper is organized as follows. First, we describe the details of MD simulations and DAE network. Then, we present results on the DAE network performance for structure and thermodynamic properties of Lennard-Jones liquids, followed by assessment of generalizability of DAE to other simple liquids. Finally, we summarize the results of this study.

**MD Simulation.** The training data for DAE is obtained through MD simulation of 12 000 standard Lennard-Jones pair potentials at various thermodynamic states (shown in Table 1) for a total simulation time of 24 µs. The Lennard-Jones potential form can be expressed as

\[ u(r) = \frac{C_6}{r^6} - \frac{C_12}{r^12} \tag{1} \]

where \( C_6 \) and \( C_12 \) are the repulsive and attractive components of the IPL pair potential with exponents \( q > p > q \). and \( A \) and \( k_B \) are the strength and inverse screening length parameters of EXP and Yukawa pair potentials (potential parameters are given in the Supporting Information Section S.2). All the MD simulations are performed using GROMACS with a time step of 1 fs in the NVT ensemble with the temperature being controlled with a time constant of 0.2 ps using the Nosé–Hoover thermostat. A total of 800 single snapshot RDFs of each system (for a total of 9.6 million single snapshots) at various times are calculated from 2 ns of simulation, where the first 200 ps is discarded. Each RDF is stored as an input vector along with its corresponding thermodynamic state, and the DAE output is the temporally averaged RDF. The temporally averaged RDF \( g_{\text{avg}}(r) \) is calculated using the expression

\[ g_{\text{avg}}(r) = \frac{1}{N} \sum_{j=1}^{N} g_j(r) \tag{5} \]

Table 1. Thermodynamic States and the Range of Parameters Used in MD Simulation of Lennard-Jones Systems

<table>
<thead>
<tr>
<th>thermodynamic state</th>
<th>Lennard-Jones potential parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \rho ) [nm(^{-3})]</td>
<td>( T ) [K]</td>
</tr>
<tr>
<td>min</td>
<td>max</td>
</tr>
<tr>
<td>8.0</td>
<td>19.4</td>
</tr>
</tbody>
</table>

where \( g_j(r) \) is the RDF at time \( t_j \) corresponding to snapshot \( j \) and \( j \) proceeds over all snapshots of the MD trajectory, i.e., \( j \in \{1, 2, ..., N\} \). Once the temporally averaged RDF is obtained, the thermodynamic properties such as pressure \( (P) \) and isothermal compressibility \( (\chi_T) \) can be calculated using the expressions

\[ P = \rho k_B T - \frac{2\pi \rho^3}{3} \int_0^{r_g} \frac{dr}{\rho} g(r)r^3 dr \tag{6} \]

\[ \chi_T = \frac{\rho}{T} \left[ 1 + 4\pi \rho \int_0^{r_g} (g(r) - 1) r^2 dr \right] \tag{7} \]

where \( T \) and \( \rho \) are the temperature and density, respectively, and \( k_B \) is the Boltzmann constant. In practice, the integration
for calculation of isothermal compressibility is done for a finite cutoff, which we set to 1.5 nm as the RDF is almost homogeneous beyond this distance.

The temporally averaged RDF of a system at a given thermodynamic state \((T, \rho)\) can also be written as the summation of a single snapshot RDF at time \(t_j\) \(g_{rj}(r; T, \rho)\) and a distance- and time-dependent fluctuation \(\epsilon_{rj}(r; T, \rho)\) expressed as

\[
g_{rj}(r; T, \rho) = g_{rj}(r; T, \rho) + \epsilon_{rj}(r; T, \rho)
\]

where \(\epsilon_{rj}(r; T, \rho)\) is the fluctuation in the RDF of a single snapshot at time \(t_j\). The DAE in this study learns to eliminate the fluctuation term in eq 8 (discussed in more detail below; see Figure 1a for the workflow of the current study).

The generalizability of the DAE network to various simple liquids with different pair potentials is also investigated in terms of RDF error9 expressed as

\[
e_{ref}(g_{ref}(r), g_{ref}(r)) = \int_0^r \frac{|g_{ref}(r) - g_{ref}(r)|r^2}{\int_0^r g_{ref}(r)r^2 dr} dr
\]

where \(g_{ref}(r)\) and \(g_{ref}(r)\) are the estimated and reference (long-time averaged MD simulation data) RDFs, respectively.

**Denoising Autoencoder Network.** Deep neural networks are generally composed of simple units known as a perceptron or node, which are stacked in width (referred to as a layer) and in depth (successive layers). Each node receives an input signal and applies a linear transformation, followed by a nonlinear activation function resulting in an output signal. A simple autoencoder network has two parts, namely, encoder and decoder, both of which are usually modeled using deep neural networks. The encoder network receives the input and maps it into a low-dimensional (usually smaller than the input dimension) space, also known as a latent space. Then, the latent space is fed into the decoder network, which maps it back to a dimension equal to the input data dimension. The main difference between a simple autoencoder and a DAE is intentional corruption of the input of DAE through noise, which, in turn, enhances the robustness of DAE compared to a simple autoencoder (schematic representation of a DAE is shown in Figure 1b).

The DAE network employed in this study learns to map a single snapshot RDF and the corresponding thermodynamic state to its temporally averaged (long-time average) RDF of an MD simulation. The DAE network is trained over 12 000 distinct Lennard-Jones liquids at various thermodynamic states with 9.6 million single snapshot RDFs. It is important to note that noise is not added to the input RDF as the inherent fluctuations in a single snapshot RDF play the role of noise in training the DAE network, i.e., each of the 800 RDFs of a given system is a noisy version of the temporally averaged RDF of the same system. Mathematically, the DAE network performs the following mapping,

\[
g_{DAE, \rho, T}(r) = DAE(g_{rj}(r); \rho, T)
\]

where \(g_{DAE, \rho, T}(r)\) and \(g_{rj}(r)\) are temporally averaged RDFs predicted by the DAE network and MD RDF at time \(t_j\) (corresponding to a specific snapshot), respectively. The parameters (weights and biases of the encoder and decoder) in the DAE network are optimized to minimize the following expectation,

\[
\min E[g_{ref, \rho, T}(r) - f_d(f_e(g_{rj}(r); T, \rho)))]
\]

where \(f_e\) and \(f_d\) are encoder and decoder functions, respectively. Both functions are selected from a flexible class of functions.
modeled using the deep neural network and can be expressed as
\[
f_i(x_{ij}) = d_j^i [W^i_j \phi_h^i (\ldots \phi_2^i [W^2_1 \phi^1_i (W^1_{i,j}x_{i,j} + b_1^i) + b_2^i] + b_3^i)]
\]
(12)
\[
y_{ij} = \phi^i(W^i_j f_i(x_{i,j}) + b^i)
\]
(13)
\[
f_{ij}(y_{ij}) = \phi^i(W^i_j \phi_h^i (\ldots \phi_2^i [W^2_1 \phi^1_i (W^1_{i,j}x_{i,j} + b_1^i) + b_2^i] + b_3^i) + b_4^i)
\]
(14)
where \(x_{ij}\) is the input vector composed of a concatenation of single snapshot RDF at time \(t_j\), temperature, and density \((x_{ij} = [g_{ij}(r), T, \rho_i])\) for the Lennard-Jones system \(i\); \(y_{ij}\) is the output of encoder, i.e., latent space \((y_{ij} = \phi^i(W^i_j f_i(x_{i,j}) + b^i))\), index \(i\) specifies a specific Lennard-Jones system in the data set \((D)\), and index \(j\) represents snapshot \(j\) RDF of system \(i\) at time \(t_j\) \((\in [T]) = (1, 2, \ldots, T_n)\) where \(n = 800\) is the total number of snapshots stored from the MD simulation of system \(i\). \(g_{ij}(r)\) is composed of RDF values at various radial distances with a bin size of 0.005 nm for the RDF interval \([0.25, 1.5]\) nm. \(\phi_h^i\) is the nonlinear activation function of layer \(k\), which receives a linear transformation of output of the layer \(k - 1\) \((W^i_{k-1,j} + b^i)\). Each layer has its own weights \(W^i_k\) and bias \(b^i\) with consistent dimensions corresponding to the output of its previous layer and its number of nodes \((W^i_k \in \mathbb{R}^{d_{k-1} \times d_k} \text{ and } b^i \in \mathbb{R}^{d_k}, \text{ where } d_k\) is the number of nodes in layer \(k)\). Optimal DAE used for obtaining the results in this study has a tanh nonlinearity function for all the layers except the decoder output layer, which has no nonlinearity.

In practice, DAE minimizes the following loss function,

\[
\epsilon_{L_{DAE}} = \epsilon_{L_r} + \epsilon_{L_{reg}}
\]
(15)
where \(\epsilon_{L_r}\) and \(\epsilon_{L_{reg}}\) are the reconstruction and regularization losses, respectively. \(\epsilon_{L_r}\) defines the match between the output of the DAE network with the ground truth (temporally averaged RDF of the MD simulation). In this study, we use the mean absolute error (between output of DAE and temporally averaged RDFs) to model the reconstruction loss, i.e.,

\[
\epsilon_{L_r} = \frac{1}{|D| |I|} \sum_{i \in [D]} \sum_{j \in [I]} \left| y_{ij} - f_{ij}(\phi^i(W^i_j f_i(x_{i,j}; T_j, \rho_i)) + b^i) \right|
\]
(16)
where \(|D|\) and \(|I|\) are the number of Lennard-Jones systems and snapshots, respectively. The regularization loss is defined as

\[
\epsilon_{L_{reg}} = \gamma \sum_h ||W^i_h||^2
\]
(17)
where \(W^i_h\) are the weights of hidden layers of both encoders and decoders in eqs 12 and 14 and \(\gamma\) is the regularization factor added for stability purposes and preventing overfitting \((\gamma \in \{0.001, 0.0005, 0.0001\})\). The adaptive moment estimation optimizer \(^{40}\) is employed to minimize the loss function \((\epsilon_{L_{DAE}})\).

In addition to the above regularization procedure, the dropout technique is also applied to DAE network layers to avoid overfitting. \(^{41}\) The dropout technique randomly drops nodes in different layers with a specific probability, therefore, training only a fraction of weights \((W)\) and biases \((b)\) of each layer at each training step. The DAE network nodes are dropped with a probability of 0.25. Various DAE architectures are trained in order to find the optimal network with satisfactory perform-
ance over both the training and validation data sets. (We divide the data set of 12,000 Lennard-Jones systems into two sets, namely, training and validation data sets with 9,600 and 2,400 systems assigned to each set, respectively. The simple liquid RDFs play the role of testing data set in this study; see the Supporting Information Section S.1.) To find optimal networks, we perform multiple trainings with various architectures. Based on the results of initial experiments, we guide our architecture selection and narrow down our search space to the three-layer networks with two different architectures, which are trained extensively. The optimal network encoder has three fully connected layers with dimensions of 250, 200, and 150, respectively. The latent space has a dimension of 100, which gets fed into the decoder with three fully connected layers each with a dimension of 150, 200, and 250, respectively.

Furthermore, we determine the sampling gain \( N_T \) as the minimum number of single snapshot RDFs needed in MD such that the average of the \( N_T \) RDFs
\[
\left\langle g_{MD}(r) \right\rangle = \frac{1}{N_T} \sum_{i=1}^{N_T} g_i(r)
\]
has a smaller error,
\[
\epsilon_{g_{MD}}(g_{MD}(r), g_{ref}(r)),
\]
compared to the DAE predicted RDF error \( \epsilon_{g_{DAE}}(g_{DAE}(r), g_{ref}(r)) \). As DAE requires only one snapshot, the sampling gain can be expressed as
\[
N_T = \min \{N; \epsilon_{g_{DAE}}(g_{MD}(r), g_{ref}(r)) < \epsilon_{g_{DAE}}(g_{DAE}(r), g_{ref}(r))\}
\]
(18)

Once the DAE network is trained and an optimal architecture is found, we examine the performance of DAE by considering two different cases. First, we investigate the performance of DAE on monatomic systems described using the Lennard-Jones pair potential. Then, we investigate its generalizability to various simple liquids.

We present a comparison of DAE with traditional approaches such as the principal component analysis and independent component analysis in the Supporting Information Section S.3. The Supporting Information also provides details on why the principal component analysis and independent component analysis are not very accurate for denoising of RDFs. The assumptions commonly used in these methods, specifically independent and identically distributed Gaussian noise, are assessed for denoising using the departure test from normality\(^{42}\) and mutual information.\(^{43,44}\) The Supporting Information S.4 provides information on how to combine principal component analysis with DAE to decrease the number of nodes and size of the DAE network.

Lennard-Jones Pair Potential. We feed single snapshot RDFs of three randomly selected Lennard-Jones systems from training and validation data sets into the DAE network (potential parameters and thermodynamic states are provided in the Supporting Information Section S.2). The predicted average RDFs are shown in Figure 2, which indicate a good match between DAE predicted and temporally averaged RDFs. The gray region in Figure 2 shows the fluctuations \( (\epsilon_t(r; \rho, T)) \) of single snapshot RDFs around the temporally averaged RDF of MD. The insets in Figure 2 show the error distribution of single snapshot RDFs and DAE prediction, i.e.,
\[
\epsilon_{g_{DAE}}(g_{MD}(r), g_{ref}(r)) \text{ and } \epsilon_{g_{DAE}}(g_{DAE}(r), g_{ref}(r)).
\]
The error distribution of DAE predicted RDF is narrower compared with the error distribution of single snapshot RDFs of MD. The distribution of \( \epsilon_{g_{DAE}}(g_{DAE}(r), g_{ref}(r)) \) has a smaller mean compared with the mean of \( \epsilon_{g_{MD}}(g_{MD}(r), g_{ref}(r)) \) distribution, indicating that the error in the DAE predicted RDF is fairly independent of the input (single snapshot RDF) and the DAE network predicts the temporally averaged RDF with good accuracy.

Figure 3. Error in RDF as a function of the number of snapshots. Error in RDF from DAE is independent of the number of snapshots. RDF error from MD decreases as the number of snapshots increases. a–c. Training data set. d–f. Validation data set (please see the Supporting Information Section S.2 for the details on the data set).
To assess the efficiency of DAE, we estimate the number of single snapshot RDFs required so that the average RDF estimated from the single snapshot RDFs is of the same accuracy as the DAE predicted RDF, i.e.,

\[ \epsilon \leq \epsilon_{DAE} \]

where \( NT \) is the sampling gain defined in eq 17. Figure 3 shows the error in RDF (computed using eq 9) as a function of the number of snapshots. The RDF error from DAE is independent of the number of snapshots, i.e., once the DAE is trained, it takes a noisy RDF as input and predicts the RDF which is close to the temporally averaged RDF. As the number of snapshots increases, the averaging gets better in MD and the RDF error decreases. Figure 3 also indicates that the number of snapshots required to reach the accuracy of DAE prediction is at least 100 and in some cases even with 800 snapshots the DAE prediction has a smaller error.

To further assess the DAE performance, we performed MD simulations on argon \( (C_{12} = 9.70 \times 10^{-6} \text{ [kJ mol}^{-1} \text{nm}^6]) \), \( C_6 = 6.22 \times 10^{-7} \text{ [kJ mol}^{-1} \text{nm}^4] \) at various thermodynamic states. The single snapshot (used as input to DAE), DAE predicted, and temporally averaged RDFs are shown in Figure 4. The error between DAE predicted and temporally averaged RDF (eq 9) is an order of magnitude smaller compared to the error between a single snapshot and temporally averaged RDFs. Table 2 shows the relative pressure and isothermal compressibility obtained from MD (averaged over 300 snapshots) and DAE (the same 300 snapshots are used as inputs and the outputs are analyzed).

<table>
<thead>
<tr>
<th>thermodynamic state</th>
<th>( P/P_{MD} )</th>
<th>( \chi_T/\chi_{T,MD} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>system</td>
<td>( \rho ) [nm(^{-3})]</td>
<td>( T ) [K]</td>
</tr>
<tr>
<td>1</td>
<td>10.0</td>
<td>300</td>
</tr>
<tr>
<td>2</td>
<td>18.0</td>
<td>300</td>
</tr>
<tr>
<td>3</td>
<td>10.0</td>
<td>390</td>
</tr>
<tr>
<td>4</td>
<td>18.0</td>
<td>390</td>
</tr>
</tbody>
</table>

\( a \) Both pressure and isothermal compressibility are normalized with their corresponding values from long-time MD simulation average.

Figure 4. Comparison between DAE predicted (shown as solid red line) and temporally averaged (shown as dashed black line) RDFs for the argon Lennard-Jones system at various thermodynamic states. Input RDF (a snapshot from MD) to DAE is shown using gray circles. a. \( T = 300 \text{ K}, \rho = 10 \text{ nm}^{-3} \). b. \( T = 300 \text{ K}, \rho = 18 \text{ nm}^{-3} \). c. \( T = 390 \text{ K}, \rho = 10 \text{ nm}^{-3} \). d. \( T = 390 \text{ K}, \rho = 18 \text{ nm}^{-3} \).
thermodynamic states is also investigated, and the results are shown in Supporting Information Section S.1.

Simple Liquid Pair Potentials. In order to assess the generalizability of the DAE network, we investigate its performance by considering various simple liquids, which are known to obey similar physics with regards to their structure and dynamics (potential parameters and thermodynamic states are provided in the Supporting Information Section S.2). We feed the single snapshot RDFs obtained from exponential, Yukawa, and two different IPLs into the DAE network. The DAE predicted and temporally averaged RDFs, shown in Figure 5, are in good agreement. The distributions of $\epsilon_{\text{eff}}(g_{\text{DAE}}(r), g_{\text{ref}}(r))$ are also similar to the Lennard-Jones systems with the peak of $\epsilon_{\text{eff}}(g_{\text{MD}}(r), g_{\text{ref}}(r))$ located further apart from zero compared to the DAE $\epsilon_{\text{eff}}(g_{\text{DAE}}(r), g_{\text{ref}}(r))$. Furthermore, similar to the Lennard-Jones system, the DAE error has a narrower distribution compared with the error distribution from MD single snapshot RDFs. These results indicate the generalizability of the DAE network to other monatomic systems, especially those belonging to the class of simple liquids. The average error between DAE predicted and longtime MD RDF is 0.001, 0.002, 0.002, and 0.002 for exponential, 10-4 IPL, 14-8 IPL, and Yukawa pair potentials, respectively. Similarly, the average errors between single snapshots and longtime MD RDF are 0.037, 0.037, 0.042, and 0.033 for exponential, 10-4 IPL, 14-8 IPL, and Yukawa pair potentials, respectively. The results again indicate that the error in DAE predicted RDFs is 1 order of magnitude smaller compared to the error in single snapshot RDFs.

A deep denoising autoencoder network is trained to obtain structural and thermodynamic properties of Lennard-Jones liquids at various thermodynamic states using a single snapshot RDF as input. The algorithm is successful not only in predicting the RDF of a Lennard-Jones pair potential, but also it is generalizable to other simple liquid pair potentials such as exponential, Yukawa, and inverse-power-law potentials. For the simple liquids considered in this work, the radial distribution functions (RDFs) predicted by the denoising autoencoder network have an order of magnitude lower RDF error compared to the errors from using single snapshot RDFs. In terms of computational efficiency, the number of snapshots required from MD simulation to obtain the accuracy of DAE predicted RDF is at least 100 snapshots, making the network highly efficient. The pressure and isothermal compressibility from DAE based RDFs are also comparable with those obtained from longtime MD simulation.

Figure 5. Assessment of the generalizability of DAE for prediction of temporally averaged RDF of various simple liquids. a. Exponential pair potential. b. IPL pair potential $(p, q) = (10, 4)$. c. IPL pair potential $(p, q) = (14, 8)$. d. Yukawa pair potential (please see Table S3 of the Supporting Information Section S.2 for the details of MD simulations). Solid red lines show the DAE predicted RDF, and dashed black lines show the temporally averaged RDF of MD simulation. Gray regions show the fluctuations in RDFs of MD snapshots around the mean (temporally averaged) RDF. The insets show the distribution of $\epsilon_{\text{eff}}(g_{\text{DAE}}(r), g_{\text{ref}}(r))$ from various single snapshots with red boxes and $\epsilon_{\text{eff}}(g_{\text{MD}}(r), g_{\text{ref}}(r))$ of single snapshots with black boxes.
The Journal of Physical Chemistry Letters

ASSOCIATED CONTENT

Supporting Information
The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jpclett.9b02820.

Additional assessment of deep denoising autoencoder network performance as well as implementation details and comparison between baseline methods such as principal component analysis and independent component analysis accompanied by an algorithm to combine the traditional methods with the denoising autoencoder developed in order to reduce the dimensions of the neural network (PDF)

AUTHOR INFORMATION

Corresponding Author
*(N.R.A.) E-mail: aluru@illinois.edu.

ORCID

Ariela Moradzadeh: 0000-0003-3355-7086

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation under Grants 1545907, 1708852, 1720633, 1720701, and 1921578. The authors acknowledge the use of Blue Waters supercomputing resources at the University of Illinois at Urbana—Champaign. Furthermore, this work partially used the Extreme Science and Engineering Discovery Environment (XSEDE) Stampede2 at the Texas Advanced Computing Center through allocation TG-CDA100010.

REFERENCES


7575

DOI: 10.1021/acs.jpclett.9b02820


