Structural Causal Model for Molecular Dynamics Simulation

Qi Liu
City University of Hong Kong

Yuanqi Du
Cornell University

Fan Feng
City University of Hong Kong

Qiwei Ye
Beijing Academy of Artificial Intelligence

Jie Fu[*]
Beijing Academy of Artificial Intelligence

Abstract

Molecular dynamics (MD) simulations describe the mechanical behaviors of molecular systems through empirical approximations of interatomic potentials. Machine learning-based approaches can improve such potentials with better transferability and generalization. Among them, graph neural networks have prevailed as they incorporate the graph structure prior while learning the interatomic interactions. Nevertheless, the simple design choices and heuristics in devising graph neural networks make them lack an explicitly interpretable component to identify the true physical interactions within the underlying system. On the other extreme, physical models can give a rather comprehensive description of a system but are hard to specify. Causal modeling lies in between these two extremes, and can provide us with more modeling flexibility. In this paper, we propose a structural causal molecular dynamics model (SCMD), the first causality-based framework to model interatomic and dynamical interactions in molecular systems by inferring causal relationships among atoms from observational data. By concerning the underlying data generation process, one can enjoy a robust and flexible MD simulation model to explicitly capture the long-range and time-dependent movement dynamics. We demonstrate the efficacy of SCMD through empirical validations on the complex molecular system (i.e., single-chain coarse-grained polymers in implicit solvent) for long-duration simulation and dynamical property prediction.

1 Introduction

Molecular dynamics (MD) simulations aim to analyze the physical movements of atoms and molecules. It has become an indispensable computational tool for studying molecular system behaviors in a variety of fields, from material science and catalysis to biological processes [Boero et al., 1998, Massobrio et al., 2015, Karplus and McCammon, 2002, Holdijk et al., 2022]. However, molecular dynamics simulation often suffers from a trade-off between accuracy and efficiency. Highly accurate simulations require resource- and time-intensive ab initio quantum-mechanical calculations, while empirical force fields model interatomic potentials by simple approximations and thus cannot guarantee accuracy and transferability [Zhang et al., 2018].

In recent years, machine learning-based approaches have been introduced to model the molecular system interactions and serve as neural potential functions for MD simulation [Unke et al., 2021]. Among them, graph neural networks (GNNs) are commonly used to learn the interactions between atoms in the system [Chmiela et al., 2017, Unke and Meuwly, 2019, Schütt et al., 2017, Gasteiger et al., 2019, 2021, Batzner et al., 2022]. Although these approaches have achieved remarkable

[*] Corresponding Author: fujie@baai.ac.cn

NeurIPS 2022 AI for Science Workshop.

performance for energy and force predictions, the heuristics for considering atomic interactions in the system are rather simple, which often only include bonded interactions or set a distance threshold for atomic interactions. In contrast, none of these challenges exists in the physics-based approaches, since the approximations of parameters are given to simulate the dynamics according to different types of interaction, atoms and bonds. However, the efficiency and transferability may be limited compared to ML-based methods.

In view of the advantages and challenges of physical and ML-based approaches, we aim to strike a better trade-off between simulation accuracy and computational efficiency for molecular dynamics through the lens of causal discovery and learning. As argued in [Schölkopf et al., 2021, Peters et al., 2017] and summarized in Table 1, causal discovery and learning aim to arrive at physical models in a data-driven manner, without manually designed parameters by the physical models, which lie in between those two extremes. Therefore, instead of heavily relying on physical functions or purely applying black-box prediction models, this work explores the identification of the causal relationship among atoms in molecular systems by only using observational data. The causal connections between two atoms exist when there is a mechanistic function between them in the physical model so that we can uncover the physical interactions solely from the observed dynamics trajectories, enabling interpretability and efficiency of the modeling simultaneously.

Specifically, we proposed SCMD, which uses the structural causal model (SCM) [Pearl, 2009, Schölkopf et al., 2021] to model the dynamical behavior of molecular dynamics. SCM can describe the data generation process of the observed molecular dynamics trajectories, and the corresponding causal relationships can be represented via the dynamic Bayesian networks (DBNs) [Pearl, 2011]. In DBNs, each node denotes an atom in the system, and the associated edges represent the causal dependencies between atoms. We can learn the DBN from data by parameterizing the connections using GNNs with recurrent designs to model the spatial and temporal dynamics. Since the molecules have different causal graphs but share the same underlying physical dynamic laws, we use an autoencoder to learn the generic dynamics laws among all molecules, enabling the zero-shot generalized capability on unseen molecules. We can infer the causal graphs via encoders and predict the future forces or positions of all atoms using the decoder. In short, SCMD can learn physics-related insights, which are equipped with the capability to generalize to unseen systems.

Table 1 compares the model attributes of our approach with ML-based and physical models. In summary, our main contributions are four-fold: *(i)*. Providing a systematic framework for leveraging causal discovery to better model the physical and causal dependencies among atoms in the MD system; *(ii)*. Developing an efficient method to learn the components of the DBNs that describes the causal model of the MD system based on the neural relation inference; *(iii)*. Presenting a more accurate and robust simulation performance on MD simulation with different simulated duration. Especially, our model can predict long-duration trajectory (350 steps) with only observable states on the initial step; and *(iv)*. Providing findings on how to leverage causality to better understand the physical and chemical systems.

2 Problem definitions and preliminaries

2.1 Problem definition

Notations The design target for the MD simulation is to model the evolution of the system of the atoms in an accurate and cost-effective manner. We use $\mathbf{x}_i = \left(\mathbf{x}_i^1, \mathbf{x}_i^2, \ldots, \mathbf{x}_i^T\right)$ and $i \in \{1, 2, \ldots, N\}$, where T and N indicate the number of total time steps and atoms in the system.

Learning target Our target is to predict future k-step movements for each atom $\hat{\mathbf{x}}^{t+1:t+k} = \left(\hat{\mathbf{x}}^{t+1}, \hat{\mathbf{x}}^{t+2}, \ldots, \hat{\mathbf{x}}^{t+k}\right)$ given the current states of the atoms $\mathbf{x}^t$[2]. The objective function is to minimize the distance between the simulated trajectories and the corresponding ground truth generated by physical simulation estimator, $D(\hat{\mathbf{x}}^{t+1:t+k}, \mathbf{x}^{t+1:t+k})$.

[2] We drop the atom index in some equations for better readability. For the time index, we use the discrete-time series representation here and define the sampling rates in the experimental sections.

Table 1: Attributes of different methods. † indicates our approach.

Methods	Statistical relation analysis	Physical-related Insights	Robustness & Generalization	Efficiency & Scalability
ML-based methods	✓	✗	✗	++
Physics-based models	✓	✓	✓	-
SCMD†	✓	✓	✓	+

2.2 Structure causal molecular dynamics (SCMD)

2.2.1 Causality and structural causal model

Causal inference and discovery tools are designed to identify the underlying causal relations among variables with the interventional or observed data [Pearl, 2009]. The structural causal model (SCM) [Pearl, 2009] and potential outcome model (POM) [Rubin, 2005] are two common causal inference frameworks. In this paper, we choose to leverage SCM to model the data generation process of the molecular dynamic system. An SCM consists of sets of endogenous ($\mathbf{X}$) and exogenous ($\mathbf{U}$) variables connected by certain functions ($\mathcal{F}$) that determine the values of the variables in $\mathbf{X}$ based on the values of the variables in $\mathbf{U}$ [Pearl, 2009]. The structure can be represented by a directed acyclic graph (DAG) $\mathcal{G} := (\mathcal{V}, E)$, encoding the direct causal dependencies (edges E) among variables (node set $\mathcal{V}$).

For an MD simulation, we assume that the nodes denote atoms and edges denote interactions among atoms. Unlike ordinary SCM modeling, which excludes time evolution, in this work, we think of the SCMs as the state abstractions of the underlying temporal process [Peters et al., 2017]. We give the SCM and data generation process of MD system below.

Data generation process of MD In the MD system, the movement of each atom will be affected by a group of atoms through physical links (*e.g.*, chemical bounds). Therefore, at time step t, the observed trajectory $\mathbf{X}^t = (\mathbf{x}_1^t, \mathbf{x}_2^t, \ldots, \mathbf{x}_N^t)$ is generated by the following dynamical process, modeled by an SCM:

$$\mathbf{x}_i^{t+1} = f_i\left(\text{PA}\left(\mathbf{x}_i^t\right)\right) + \mathbf{u}_i^t \tag{1}$$

where $\text{PA}\left(\mathbf{x}_i^t\right) = \left\{\mathbf{x}_j^{t-1} | \forall x_j = \text{PA}\left(x_i\right) \text{ in } \mathcal{G}\right\}$, where $\mathcal{G}$ is the skeleton graph of the SCM. The nodes are denoted by a set of atom indexes $\{x_1, x_2, \ldots, x_N\}$ the corresponding adjacent matrix $\mathcal{A}$ can be estimated by the observed data. We assume that the movement of the atoms at $t+1$ is only affected by those at time step t, similarly to those works on causal dynamic systems [Lippe et al., 2022, Wang et al., 2022, Löwe et al., 2022]. In other words, there are no instantaneous temporal effects in the SCM. Fig. 1(a) indicates the data generation process, including the observed trajectories and the underlying SCM. Notably, we assume that there exists a self-loop for all atoms. This is because the movements of each atom are affected by their historical movement variables (*e.g.*, locations). Hence, the SCM (Fig. 1(a)) may contain self-loops and cycles in the graph. However, this will not affect the intervention and identifiability theory of causal discovery since the time-dependent process (*i.e.*, full-time graph) does not have cycles (Fig. 1(b)) [Peters et al., 2017]. To conclude, the SCM can extract causal relations among different atoms, expressing the underlying data generation process that takes place over time.

2.2.2 Dynamic Bayesian networks

A Bayesian network (BN) [Dean and Kanazawa, 1989] over the set $\mathcal{X}$ is defined as $\mathcal{B} = (\mathcal{G}_{\mathcal{X}}, P)$, where $\mathcal{G}_{\mathcal{X}}$ is the graph structure of the network, giving the dependency connectivity among the variables, and the distribution P gives the conditional probabilities of the variables in $\mathcal{X}$:

$$P(\mathcal{X}) = \prod_{X_i \in \mathcal{X}} P_i\left(X_i \mid \text{PA}\left(X_i\right)\right) \tag{2}$$

DBNs [Murphy, 2002] are one of the extended versions of BN, modeling the time-dependent process of variables and their conditional dependence. The unfolded DBNs can be represented as BNs. The variables in DBNs are in discrete time steps and depend on variables only from the same and previous time slices. The network structure and conditional dependence are the same as in other time

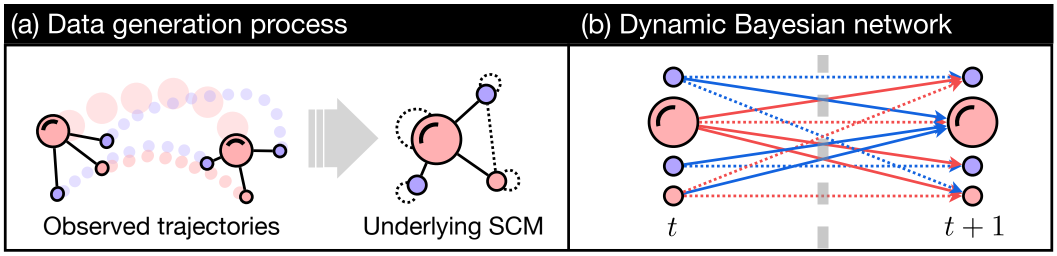

Figure 1: Conceptual illustrations of the observed trajectories, the corresponding SCM, and the DBN, where the solid lines and dashed lines indicate the different causal edge types: physically bounded and non-bounded, respectively. For brevity, we ignore the edge directions in the underlying SCM. Best viewed in color.

steps, which assume the stationary property of the system. We can infer the SCM by approximating the distribution over the DBN in Equation 2. In other words, we use DBN to model the SCM in a probabilistic way.

To model the probabilistic SCM with temporal effects, each conditional dependence in the DBN should represent the causal relationship. We assume $\mathcal{B}$ is the DBN model over $\{x_1^1, x_2^1, \ldots, x_{N-1}^t, x_N^t\}$. Thus, the DBN is the unfolded causal graph with connections beyond time steps, modeling the temporal effects among atom pairs. Fig. 1(b) displays the visualized DBN with two-time slices. In the present paper, we learn the DBN using the auto-encoder model, similar to [Löwe et al., 2022], where the causal dependence structure is inferred via the latent dimensions, and the temporal dynamics are estimated by the decoder.

2.2.3 Neural relation inference

We use the neural relation inference (NRI) framework [Kipf et al., 2018, Löwe et al., 2022] to learn the DBN. In NRI, the interactions among entities are learned via the variational auto-encoder (VAE) model, where the interactions are modeled as the latent codes inferred by the encoder, and the decoder reconstructs the original trajectories. The evidence lower bound (ELBO) of the VAE is given by:

$$\mathcal{L}(\phi, \theta) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{X})}\left[\log p_\theta(\mathbf{X} \mid \mathbf{z})\right] - \mathrm{KL}\left[q_\phi(\mathbf{z} \mid \mathbf{X}) \| p(\mathbf{z})\right], \tag{3}$$

where, the latent vector $\mathbf{z}$ denotes the inferred interaction among entities $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ and we have observed trajectories $\mathbf{x}_i = \left(\mathbf{x}_i^1, \ldots, \mathbf{x}_i^T\right)$ of all entities. N and T are the number of entities and time steps, respectively. $p\left(\mathbf{z}\right)$ is the prior distribution of the latent space. ϕ and θ denote the encoder and decoder, respectively.

The amortized causal discovery [Löwe et al., 2022] further proposes to leverage the latent space to learn causal relations between entities. Specifically, the work extends the NRI with an additional edge type defining the non-causal relations in the latent space.

In our work, we follow the designed structure of the amortized causal discovery, where we learn the DBN using the VAE model and estimate the temporal dynamics by the decoder. In the original amortized causal discovery, the applied scenarios are relatively simple and the datasets are all with low dimensions (*e.g.*, 1-D time-series, particles, and simulated fMRI data). We explore how to leverage the framework to model the complex MD system with possible noisy and high-dimensional data. In particular, Zhu et al. [2022] also utilize the NRI framework to infer the latent interactions of protein allosteric communications. Different from [Zhu et al., 2022], we aim to propose a generic framework for discovering the causal links in a molecular dynamics system.

3 Structural Causal Molecular Dynamics

We utilize a generative model, named sequential causal VAE (SC-VAE), to learn the graph structure and temporal causal dependencies of DBN in an end-to-end fashion. The overall pipeline is illustrated in Fig. 2. In SC-VAE, the atom representation is learned via several GNN layers and serves as the input of the network. The encoder infers the distribution of causal graphs given the atom representation. Specifically, we model the adjacent matrix of the causal graph as the latent factors. Hence, the produced causal graphs are atom-specific, making the framework generic and flexible. The decoder generates future movements by modeling the temporal interactions among atoms with the sampled causal graph instance from the latent space and the learned atom representation. The supervision signals include the sequential prediction error and the KL-divergence between the posterior and prior distribution of the causal graph. We discuss each component below in our proposed framework.

3.1 Learning the generative process of SCMD

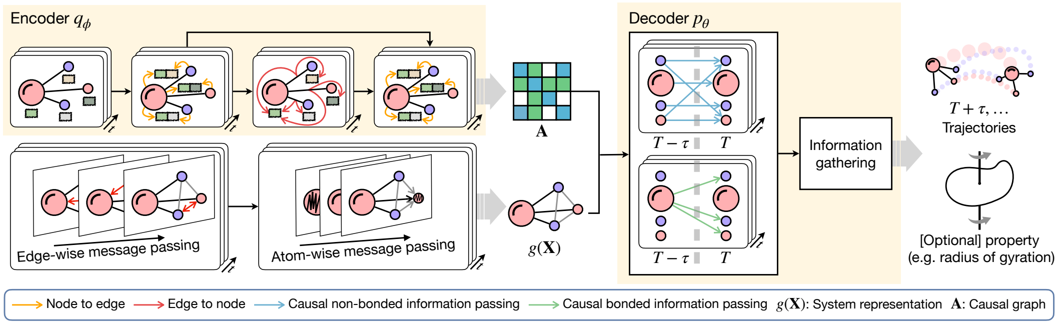
Encoder q_ϕ
Decoder p_θ
$\mathbf{A}$
$g(\mathbf{X})$
Edge-wise message passing
Atom-wise message passing
$T-\tau$
T
$T-\tau$
T
Information gathering
$T+\tau, \ldots$
Trajectories
[Optional] property
(e.g. radius of gyration)
Node to edge
Edge to node
Causal non-bonded information passing
Causal bonded information passing
$g(\mathbf{X})$: System representation
$\mathbf{A}$: Causal graph

Figure 2: Overview of learning framework SC-VAE. The square below the atom in the encoder part indicates atom representation. And the concatenation of two squares indicates edge representation.

Learning the edge representation $\mathcal{H}$ Following NRI, we use the neural message passing to obtain the edge embeddings as the input to estimate the causal dependencies. The edges embeddings $\mathcal{H}$ are computed based on the message passing on a fully-connected graph, where each atom will propagate the message to all the remaining in the molecular system. Thus, the edge embedding is $\mathcal{H} = \{\forall i, j \in \{1, \ldots, N\}$ and $i \neq j | \mathbf{h}(\mathbf{x}_i, \mathbf{x}_j)\}$, where $\mathbf{h}(\mathbf{x}_i, \mathbf{x}_j)$ is the embedding of edge connected with $\mathbf{x}_i$ and $\mathbf{x}_j$. Specifically, to calculate $\mathbf{h}(\mathbf{x}_i, \mathbf{x}_j)$, we follow the steps below:
Step 1: Atom-wise message passing

$$\mathbf{h}_0\left(\mathbf{x}_i, \mathbf{x}_j\right) = f_{e1}\left(f_{\text{emb}}(\mathbf{x}_i), f_{\text{emb}}(\mathbf{x}_j)\right),$$

where f_{emb} is composed of several stacking MLP layers, deriving the initial embedding representation of atoms. f_{e1} is to compute the message passing between atom pairs, implemented by MLP layers. The output of step 1 is the initial edge embedding $\mathbf{h}_0$.
Step 2: Edge-wise message passing

$$\mathbf{f}_j = f_v\left(\sum_{i \neq j} \mathbf{h}_0(\mathbf{x}_i, \mathbf{x}_j)\right),$$

where f_v is the edge message passing function, implemented by the stacking MLP layers. $\mathbf{f}_j$ is the updated atom representation with the messages passed from all the edges.
Step 3: Final Edge embedding

$$\mathbf{h}\left(\mathbf{x}_i, \mathbf{x}_j\right) = f_{e2}\left(\mathbf{f}_i, \mathbf{f}_j\right),$$

where f_{e2} is another module to compute the message passing between atoms, similar to f_{e1} but with different parameters. The final edge embedding between $\mathbf{x}_i$ and $\mathbf{x}_j$ is then derived as $\mathbf{h}\left(\mathbf{x}_i, \mathbf{x}_j\right)$.

The edge embeddings in $\mathcal{H}$, are then used to estimate the distribution of the causal graph and the temporal dynamics.

Learning the causal graph structure A Following the causal discovery with NRI framework [Kipf et al., 2018, Löwe et al., 2022], we utilize an encoder to learn the sample-agnostic causal graph. This is particularly appropriate because most molecule samples with the same chemical space share the same underlying physical dynamics given some sound conditions (*e.g.*, same media), while the causal graphs for describing the atom movements are sample-specific due to the different molecular structures. The encoder q_ϕ, hence, to learn the structure of the causal graphs among all samples. Given a trajectory data sample $\mathbf{x}_t$ at time step t, the encoder output the distribution of the adjacent matrix of the sample $q_\phi\left(\mathbf{A}|g\left(\mathbf{x}_t\right)\right)$, where $\mathbf{A}$ is the adjacent matrix and q_ϕ is parameterized via several dense layers empirically.

We have three edge types, indicating different causal dependencies below:
Type I: Non-causal edges a_0;
Type II: Causal edges with bounded links a_1;
Type III: Causal edges without bounded links a_2.

Hence, $q_\phi\left(\mathbf{A}|g\left(\mathbf{x}_t\right)\right)$ will generate the probability distribution of each type edge based on the embedding and sparse constraints.

$$q_\phi\left(\boldsymbol{a}_{ij} \mid g\left(\mathbf{x}_i, \mathbf{x}_j\right)\right) = \operatorname{softmax}\left(f_{\mathrm{c}}(\mathbf{h}\left(\mathbf{x}_i, \mathbf{x}_j\right))\right),$$

where f_{c} is a dense layer to classify the edge type. $\boldsymbol{a}_{ij} \in \mathbb{R}^3$, indicating the likelihoods of which types of casual links the edge belong to. We have the prior beliefs that the edge types are uniformly distributed. Hence, the prior distribution of $p(\mathbf{A})$ is a uniform distribution and $\mathcal{L}_{\mathrm{KL}} = \mathrm{KL}(q_\phi(\mathbf{A}|\mathbf{x})\|p(\mathbf{A}))$. In the meantime, we also have the regularization terms based on the sparse causal mechanism [Schölkopf et al., 2021] of $\mathbf{A}$:

$$\mathcal{L}_{\mathrm{reg}} = \|p(a_1) + p(a_2)\|_1$$

$p(a_1)$ and $p(a_2)$ correspond to the total output likelihoods of causal edges of all atoms.

Learning the temporal dynamic model p_θ With the learned causal structure $\mathbf{A}$, we have the skeleton of the DBN and the only thing to do is estimating the edge strength of each path in DBN. Formally, the path strength links to the dynamics model $p_\theta\left(\mathbf{x}_{t+1}|g\left(\mathbf{x}_t\right), \mathcal{A}\right)$. We use g to represent all functions involved to compute the edge embeddings $\mathcal{H}$. We utilize two GNNs to parameterize p_θ. Specifically, one GNN f_{dyn1} is trained to model the interaction between those atoms with edge Type III, and another f_{dyn2} is to model those atoms linked with edge Type II. The loss for the temporal dynamic prediction is $\mathcal{L}_{\mathrm{dyn}} = \mathrm{MSE}\left(\mathbf{x}_{t+1}, p_\theta\left(\mathbf{x}_{t+1}|g\left(\mathbf{x}_t\right), \mathcal{A}\right)\right)$, where MSE denotes the mean square error.

Total loss In general, the total loss is given below: $\mathcal{L}_{\mathrm{total}} = \alpha_1\mathcal{L}_{\mathrm{dyn}} + \alpha_2\mathcal{L}_{\mathrm{KL}} + \alpha_3\mathcal{L}_{\mathrm{reg}}$, where $\{\alpha_1, \alpha_2, \alpha_3\}$ are tunable hyper-parameters.

4 Experiments

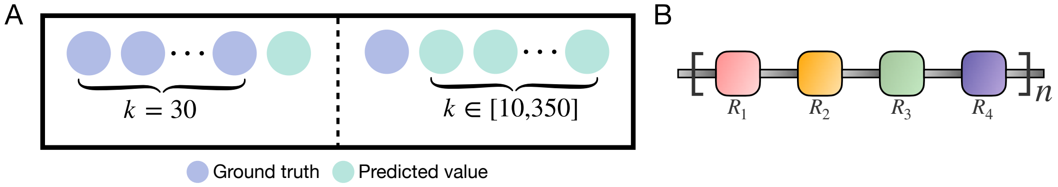

Figure 3: **(A)** The illustrative MD trajectory prediction. The left side is the prediction method from MLCGMD, which uses previous 30 historical movement information for next-timestep output. The right side is our prediction method, which uses only 1 previous movement information for a sequence of k-timesteps outputs. $k \in [10, 350]$. **(B)** Polymer structures for training and testing, where each polymer is composed of 4 types of CUs (best viewed in color).

4.1 Setup

We validate the effectiveness of our method on long-range and long-duration MD simulations. Specifically, we aim to simulate $\mathbf{x}^{2:T}$ by only observing the $\mathbf{x}^1$ of the molecular system.

Training To achieve this, during the training phase, we have the collected MD trajectories $\mathcal{D}_{\mathrm{tr}}$, each trajectory corresponds to a specific molecule with the number of time steps T. We train the SC-VAE using $\mathcal{D}_{\mathrm{tr}}$ and fix the trained parameters q_ϕ and p_θ. We observe the initial states (*i.e.*, locations and velocities) $\mathbf{x}^1$ of all atoms and predict the future steps with the inferred causal relations and temporal dynamics. At $t = 2$, we use the initial value $\mathbf{x}_1$ of the system as input. To predict the states of atoms at time step t ($t > 2$), we use the estimated states $\hat{\mathbf{x}}^{t-1}$ at time step $t - 1$.

Simulation During the simulation phase, we use the trained model to predict the whole trajectories by observing the initial states (see the illustration at the r.h.s. of Fig. 3(A)).

Dataset We validate the performance of SC-VAE over a single-chain coarse-grained polymer dataset simulated with LAMMPS [Thompson et al., 2022]. The simulation setup, such as force field parameters, is described in [Webb et al., 2020] and the dataset is simulated by [Fu et al., 2022]. As shown in Fig. 3(B), each polymer is composed of 4 types of constitutional units (CUs) with different numbers (n) of the such repeated pattern. And the dynamics of each polymer corresponds to the trajectory of 50k τ timesteps with a recording frequency of 5τ, where τ is the unit time. We have a total of 100 polymers, 90 of which are used as the training set, and the remaining 10 polymers are used for simulation.

Baseline We compare our method with MLCGMD [Fu et al., 2022], the first work exploring the coarse-graining simulation of MD using ML tools. MLCGMD utilizes a score-based GNN to model atomic interactions, which embeds and coarse-grains an MD system to a coarse-level graph via clustering. At time step t, the historical states $\mathbf{x}^{t-k:t-1}$ are treated as the input and the framework generates the simulated state at time t (see the l.h.s. of Fig. 3(A)).

4.1.1 Implementation

Training and simulation Our model generates the simulated trajectory with a duration of T steps. That is, we split the whole trajectory into $\lfloor 10,000/T \rfloor$ disjoint sequences and predict the atom movements in each sequence by observing the first state. To test the robustness of our method in terms of the sequence length, we vary $T \in \{10, 50, 100, 150, 200, 250, 300, 350\}$. Different from ours, MLCGMD utilizes the input with 30 history states, and generates the next step each time.

Experimental details The hyper-parameters we used are summarized in Appendix Table B and C. All methods are implemented using PyTorch [Paszke et al., 2019] toolbox with an Intel Core i9 CPU and 4 Nvidia RTX 3090 GPUs.

4.2 Results

4.2.1 Evaluation metrics

Following [Fu et al., 2022], we use the estimated surrogate property of the radius of gyration (R_g^2) to evaluate the precision of the simulation models. We use two metrics in terms of R_g^2:

Earth mover's distance Earth mover's distance (EMD) is a measure of the distance between two probability distributions over a specific region. Here, we use EMD to measure the distance between simulated trajectories and the ground truth. Hence, a small EMD leads to a more accurate simulation.

Mean absolute error Mean absolute error (MAE) is also adopted to measure the distance between the simulated movements and the ground truth for each time step.

4.2.2 Results and analysis

Fig. 4 gives the comparison between our method (with $T = 350$ and 10) and MLCGMD, where the plots indicate the simulated R_g^2 for all time steps. From the plots, we can find our model leads to a more accurate dynamic simulation. We give a detailed analysis in terms of the robustness of the simulation duration and sample efficiency below.

Robustness against different simulation duration T. We vary the simulation duration $T \in \{10, 50, 100, 150, 200, 250, 300, 350\}$. The full results are given in Appendix Fig. A. Though the accuracy is higher when the duration is smaller, our model can still result in a better simulation than the baseline when the duration is extremely long (*e.g.*, $T = 30$). The results verify that our model

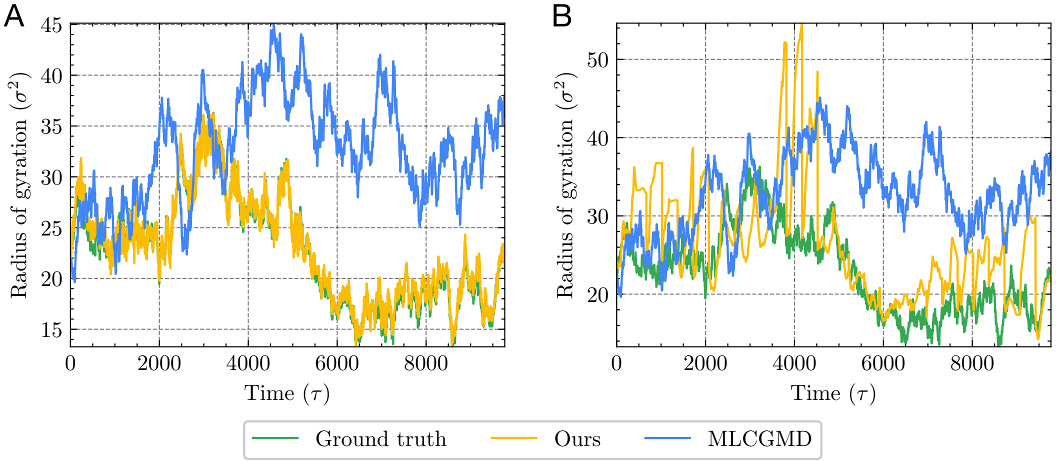

Figure 4: R_g^2 prediction performance for our method and MLCGMD. (**A**) The simulated duration is 10τ; (**B**) The simulated duration is 350τ.

can predict a very long sequence with only one truth observation of the initial step. However, the baseline model requires 30 steps of historical values to predict a future step. This makes our model robust and potentially be applied to many down-streaming MD applications (e.g., protein modeling).

Better simulation with fewer samples. Table A gives the results of accuracies versus the required samples. We follow the original settings in MLCGMD to use only 30 samples at the beginning of the trajectories to train the model. We find that our model can learn a better simulation model (lower MAE and EMD) with fewer samples (28 versus 30) compared with MLCGMD.

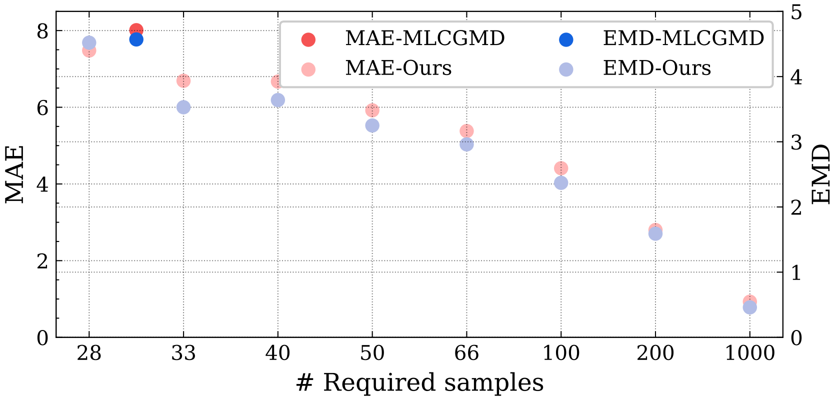

Figure 5: Results on MAE and MED of radius of gyration versus the number of samples required for simulation. Quantitative results are given in Appendix Table A.

5 Conclusions

We propose the Structural Causal Molecular Dynamics (SCMD), exploring the potential to take advantage of causality to learn a more accurate and robust molecular dynamic simulation. We formalize the MD system as a structural causal model and utilize a dynamic Bayesian network to estimate the causal model. The causal model does not model the specific functions of MD, but instead includes the interaction relations and strength analogously to the true physical dynamics. To learn the model, we apply and extend the neural relation inference framework to the MD system to learn the DBN in an end-to-end fashion. The empirical results verify the effectiveness and efficiency of our model with different prediction durations. Based on both algorithmic and empirical aspects, we believe that our model can be applied to many downstream MD applications, e.g., drug discovery.

Acknowledgement

We thank Ge Zhang, Yiming Liang, Zekun Wang for helping with experiments, and Yucheng Zhang, Chenqing Hua, Sitao Luan, Shuangjia Zheng, Jianmin Wang for helpful discussions.

References

Mauro Boero, Michele Parrinello, and Kiyoyuki Terakura. First principles molecular dynamics study of ziegler- natta heterogeneous catalysis. *Journal of the American Chemical Society*, 120 (12):2746–2752, 1998.

Carlo Massobrio, Jincheng Du, Marco Bernasconi, and Philip S Salmon. Molecular dynamics simulations of disordered materials. *Cham: Springer International Publishing*, 2015.

Martin Karplus and J Andrew McCammon. Molecular dynamics simulations of biomolecules. *Nature structural biology*, 9(9):646–652, 2002.

Lars Holdijk, Yuanqi Du, Ferry Hooft, Priyank Jaini, Bernd Ensing, and Max Welling. Path integral stochastic optimal control for sampling transition paths. *arXiv preprint arXiv:2207.02149*, 2022.

Linfeng Zhang, Jiequn Han, Han Wang, Roberto Car, and EJPRL Weinan. Deep potential molecular dynamics: a scalable model with the accuracy of quantum mechanics. *Physical review letters*, 120(14):143001, 2018.

Oliver T Unke, Stefan Chmiela, Huziel E Sauceda, Michael Gastegger, Igor Poltavsky, Kristof T Schutt, Alexandre Tkatchenko, and Klaus-Robert Muller. Machine learning force fields. *Chemical Reviews*, 121(16):10142–10186, 2021.

Stefan Chmiela, Alexandre Tkatchenko, Huziel E Sauceda, Igor Poltavsky, Kristof T Schutt, and Klaus-Robert Muller. Machine learning of accurate energy-conserving molecular force fields. *Science advances*, 3(5):e1603015, 2017.

Oliver T Unke and Markus Meuwly. Physnet: A neural network for predicting energies, forces, dipole moments, and partial charges. *Journal of chemical theory and computation*, 15(6):3678–3693, 2019.

Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *Advances in neural information processing systems (NeurIPS)*, 30, 2017.

Johannes Gasteiger, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs. In *International Conference on Learning Representations (ICLR)*, 2019.

Johannes Gasteiger, Florian Becker, and Stephan Günnemann. Gemnet: Universal directional graph neural networks for molecules. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:6790–6802, 2021.

Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature communications*, 13(1): 1–11, 2022.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

Judea Pearl. Causal inference in statistics: An overview. *Statistics surveys*, 3:96–146, 2009.

Judea Pearl. Bayesian networks. 2011.

Donald B Rubin. Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, 100(469):322–331, 2005.

Phillip Lippe, Sara Magliacane, Sindy Löwe, Yuki M Asano, Taco Cohen, and Stratis Gavves. Citris: Causal identifiability from temporal intervened sequences. In *International Conference on Machine Learning (ICML)*, pages 13557–13603. PMLR, 2022.

Zizhao Wang, Xuesu Xiao, Zifan Xu, Yuke Zhu, and Peter Stone. Causal dynamics learning for task-independent state abstraction. In *International Conference on Machine Learning (ICML)*, pages 23151–23180. PMLR, 2022.

Sindy Löwe, David Madras, Richard Zemel, and Max Welling. Amortized causal discovery: Learning to infer causal graphs from time-series data. In *Conference on Causal Learning and Reasoning (CLeaR)*, pages 509–525. PMLR, 2022.

Thomas Dean and Keiji Kanazawa. A model for reasoning about persistence and causation. *Computational intelligence*, 5(2):142–150, 1989.

Kevin Patrick Murphy. *Dynamic bayesian networks: representation, inference and learning*. University of California, Berkeley, 2002.

Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *International Conference on Machine Learning (ICML)*, pages 2688–2697. PMLR, 2018.

Jingxuan Zhu, Juexin Wang, Weiwei Han, and Dong Xu. Neural relational inference to learn long-range allosteric interactions in proteins from molecular dynamics simulations. *Nature communications*, 13(1):1–16, 2022.

Aidan P Thompson, H Metin Aktulga, Richard Berger, Dan S Bolintineanu, W Michael Brown, Paul S Crozier, Pieter J in't Veld, Axel Kohlmeyer, Stan G Moore, Trung Dac Nguyen, et al. Lammps-a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Computer Physics Communications*, 271:108171, 2022.

Michael A Webb, Nicholas E Jackson, Phwey S Gil, and Juan J de Pablo. Targeted sequence design within the coarse-grained polymer genome. *Science advances*, 6(43):eabc6216, 2020.

Xiang Fu, Tian Xie, Nathan J. Rebello, Bradley Olsen, and Tommi S. Jaakkola. Simulate time-integrated coarse-grained molecular dynamics with geometric machine learning. In *ICLR Workshop on Deep Generative Models for Highly Structured Data*, 2022.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems (NeurIPS)*, 32, 2019.

Zizhao Wang, Xuesu Xiao, Yuke Zhu, and Peter Stone. Task-independent causal state abstraction. *NeurIPS 2021 Workshop on Robot Learning: Self-Supervised and Lifelong Learning*, 2021.

Frank Noé, Alexandre Tkatchenko, Klaus-Robert Müller, and Cecilia Clementi. Machine learning for molecular simulation. *Annual review of physical chemistry*, 71:361–390, 2020.

Justin S Smith, Olexandr Isayev, and Adrian E Roitberg. Ani-1: an extensible neural network potential with dft accuracy at force field computational cost. *Chemical science*, 8(4):3192–3203, 2017.

Stefan Doerr, Maciej Majewski, Adrià Pérez, Andreas Kramer, Cecilia Clementi, Frank Noe, Toni Giorgino, and Gianni De Fabritiis. Torchmd: A deep learning framework for molecular simulations. *Journal of chemical theory and computation*, 17(4):2355–2363, 2021.

Frank Noé, Simon Olsson, Jonas Köhler, and Hao Wu. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457):eaaw1147, 2019.

Oliver Nelles. Nonlinear dynamic system identification. In *Nonlinear System Identification*, pages 547–577. Springer, 2001.

Alvaro Sanchez-Gonzalez, Nicolas Heess, Jost Tobias Springenberg, Josh Merel, Martin Riedmiller, Raia Hadsell, and Peter Battaglia. Graph networks as learnable physics engines for inference and control. In *International Conference on Machine Learning (ICML)*, pages 4470–4479. PMLR, 2018.

Eugene M Izhikevich. *Dynamical systems in neuroscience*. MIT press, 2007.

Jay J Van Bavel, Oriel FeldmanHall, and Peter Mende-Siedlecki. The neuroscience of moral cognition: from dual processes to dynamic systems. *Current Opinion in Psychology*, 6:167–172, 2015.

Jens Kober and Jan Peters. Learning motor primitives for robotics. In *2009 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2112–2118. IEEE, 2009.

Benoît Girard, Nicolas Tabareau, Quang-Cuong Pham, Alain Berthoz, and J-J Slotine. Where neuroscience and dynamic system theory meet autonomous robotics: a contracting basal ganglia model for action selection. *Neural Networks*, 21(4):628–641, 2008.

Stephan Bongers, Tineke Blom, and Joris M Mooij. Causal modeling of dynamical systems. *arXiv preprint arXiv:1803.08784*, 2018.

Joris M Mooij, Dominik Janzing, and Bernhard Schölkopf. From ordinary differential equations to structural causal models: the deterministic case. *arXiv preprint arXiv:1304.7920*, 2013.

Tineke Blom and Joris M Mooij. Generalized structural causal models. *arXiv preprint arXiv:1805.06539*, 2018.

Jonas Peters, Stefan Bauer, and Niklas Pfister. Causal models for dynamical systems. In *Probabilistic and Causal Inference: The Works of Judea Pearl*, pages 671–690. 2022.

Weiran Yao, Guangyi Chen, and Kun Zhang. Temporally disentangled representation learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Sebastian Weichwald, Søren Wengel Mogensen, Tabitha Edith Lee, Dominik Baumann, Oliver Kroemer, Isabelle Guyon, Sebastian Trimpe, Jonas Peters, and Niklas Pfister. Learning by doing: Controlling a dynamical system using causality, control, and reinforcement learning. *arXiv preprint arXiv:2202.06052*, 2022.

Biwei Huang, Fan Feng, Chaochao Lu, Sara Magliacane, and Kun Zhang. Adarl: What, where, and how to adapt in transfer reinforcement learning. *International conference on learning representations (ICLR)*, 2021.

Fan Feng, Biwei Huang, Kun Zhang, and Sara Magliacane. Factored adaptation for non-stationary reinforcement learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Appendix for Structural Causal Models for Molecular Dynamics Simulations

A Limitations and future works

In this work, we only use the sparse constraints to learn the causal graph from the observational data. Though it is a normal setup in many works on causal representation learning for dynamic systems [Wang et al., 2021], it lacks some grounded causal discovery foundations. In future work, we plan to explore more aspects of causality, including intervention and counterfactual learning [Peters et al., 2017], to learn the causal graph and its applicability to complex MD simulation systems.

B Related work

B.1 Machine Learning for Molecular dynamics

Owing to its efficiency and transferrability, machine learning methods have been extensively studied for molecular dynamics [Noé et al., 2020]. Specifically, there are two main ways that machine learning methods have been used to improve the accuracy and/or efficiency of molecular dynamics simulations. The first way is to leverage machine learning models to fit hand-crafted parameters in molecular dynamics simulations. For example, traditional molecular dynamics rely heavily on hand-crafted force field parameters. Machine learning methods are used to learn from ab initio quantum-mechanical data and serve as a machine learning force field for molecular dynamics with ab initio-level accuracy and increased efficiency [Smith et al., 2017, Zhang et al., 2018, Doerr et al., 2021, Batzner et al., 2022]. Another line of work leverages deep generative models to learn to sample probability distribution from the observational data. Boltzmann generator proposes a normalizing flow that generates samples from the equilibrium distribution of the molecular system [Noé et al., 2019]. This approach can be used to avoid incredibly long molecular dynamics simulations for studying complex dynamical behaviors such as protein folding.

B.2 Causal Dynamical System

Dynamical systems have been widely used to describe the dynamical behaviors of systems in various fields, including physical systems [Nelles, 2001, Sanchez-Gonzalez et al., 2018, Nelles, 2001], neuroscience [Izhikevich, 2007, Van Bavel et al., 2015], robotics [Kober and Peters, 2009, Girard et al., 2008]. Recently, a range of works in the causality literature aim to uncover the causal semantics of the dynamic system. Bongers et al. [2018] provide the formal framework named structural dynamical causal models (SDCMs) to model the dynamics system as the structured casual model with differential equations. Mooij et al. [2013], Blom and Mooij [2018] describe the asymptotic solution of the dynamic systems as a causal model and consider the interventions on ODE systems. Peters et al. [2022] propose causal kinetic models to model the continuous time models with ODE functions by intervening on the dynamic systems. Based on these theoretical frameworks, several works have been proposed to learn the causal representation of dynamic systems under real-world or simulated scenarios. Yao et al. [2022] learn latent causal dynamic systems under modular distribution shifts and provide the conditions for identifiability guarantees on causal graphs. Lippe et al. [2022] learn the causal factors from the sequences of observations and design interventions on dynamic systems. Weichwald et al. [2022] propose a general framework to use causality for the control tasks in dynamic systems. Huang et al. [2021], Feng et al. [2022] learn the factored and causal structure of the dynamic systems in RL under heterogeneous and non-stationary environments. In this work, we first leverage the causal dynamic system to model molecular dynamics, learning how the atoms evolve and interact temporally in the system.

C Full results and hyper-parameters

Fig. A and Table A give the full results with different simulated durations and required label samples. Table B and C list the chosen architectures and hyper-parameters, respectively.

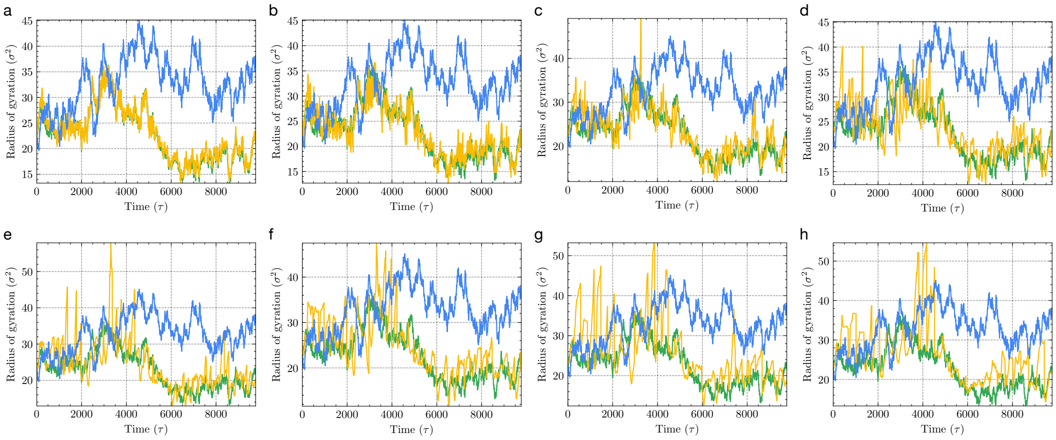

Figure A: R_g^2 prediction performance for our method and MLCGMD. The simulation duration is 10τ (a), 50τ (b), 100τ (c), 150τ (d), 200τ (e), 250τ (f), 300τ (g), and 350τ (h), respectively.

	MLCGMD [Fu et al., 2022]	Ours							
# Required samples	30	28	33	40	50	66	100	200	1000
T	–	350	300	250	200	150	100	50	10
MAE	8.01	7.48	6.69	6.67	5.92	5.38	4.41	2.80	0.93
EMD	4.57	4.52	3.53	3.64	3.25	2.96	2.37	1.59	0.46

Table A: The results on MAE and MED of radius of gyration (R_g^2) versus different number of samples and simulation duration T.

	Layer	# Nodes	Activation function
f_{enc}	MLPs (atom)	256 - 256	ELU - ELU + BatchNorm
	MLP (edge)	256 - 256	ELU - ELU + BatchNorm
	MLP (atom)	256 - 256	ELU - ELU + BatchNorm
	MLP (edge)	256 - 256	ELU - ELU + BatchNorm
f_{dec}	MLP (static atom)	64	ReLU+LayerNorm
	MLP (static edge)	64	ReLU+LayerNorm
	MLP (static edge)	64	ReLU+LayerNorm
	MLP (static node)	64	ReLU+LayerNorm
	MLP (static edge)	64 - 64	ReLU+LayerNorm - ReLU+LayerNorm
	MLP (static node)	64 - 64	ReLU+LayerNorm - ReLU+LayerNorm
	MLP (temporal atom)	128	ReLU+LayerNorm
	MLP (temporal edge)	128	ReLU+LayerNorm
	MLP (temporal edge)	[128 - 128] $\times 2$	[ReLU+LayerNorm - ReLU+LayerNorm] $\times 2$
	MLP (temporal node)	128 - 128	ReLU+LayerNorm - N/A

Table B: The architecture for both the encoder and decoder.

Hyperparameter	**Value**
Batch size	32
Optimizer	Adam Kingma and Ba [2014]
Learning rate	$1e-4$
Learning rate scheduler	0.1
Number of epochs	14
α	$\{1, 1e-2, 1e-2\}$

Table C: An overview of the hyperparameter used.