Hydrogen Diffusion through Polymer using Deep Reinforcement Learning

Tian Sang ∗
University of Southern California
tians@usc.edu

Ken-ichi Nomura ∗
University of Southern California
knomura@usc.edu

Aiichiro Nakano ∗
University of Southern California
anakano@usc.edu

Rajiv K. Kalia ∗
University of Southern California
rkalia@usc.edu

Priya Vashishta ∗
University of Southern California
priyav@usc.edu

Abstract

Robust and cost-effective hydrogen storage is considered as an enabling technology for carbon-free and renewable energy society. Hydrogen tank using polymer liner has been in market and already used in fuel cell electric vehicles and airplanes. Understanding of the fundamental mechanisms of hydrogen diffusion in polymer could greatly speed up the deployment of hydrogen energy infrastructure at scale. A computational framework that provides atomistic diffusion pathways at experimentally relevant time scale is ideal for this purpose, however, it is yet to be demonstrated. We have developed a novel deep reinforcement learning framework combined with transition state theory to efficiently identify molecular diffusion pathways in polymeric materials. Employing distributed replay buffer, an ensemble of agents quickly learns the complex energy landscape of the system of interest. Subsequently, the diffusion time of each pathway is estimated using transition state theory. With the distributed training framework we have achieved significant improvement in learning in terms of both the training metrics as well as the molecular diffusion time.

1 Introduction

Hydrogen energy plays an essential role in producing clean and sustainable power. To date, a variety of storage methods have been developed. They are often categorized into physical storage and chemical storage. The physical hydrogen storage methods mainly focus on storing hydrogen gas as a condensed form including compressed hydrogen gas, liquid hydrogen storage, adsorption onto materials, and others [Usman 2022]. On the other hand, reaction-based storage can provide safer transportation and reversible storage although the releasing hydrogen from chemical compounds may require additional energy input, and lower release rates. Recently Type-IV hydrogen tank with polymer liner has been attracting great attentions. High density polyethylene (HDPE) [Fujiiwara et al. 2021] and polyamide (PA) [Yersak et al. 2017] are widely used materials for the liner due to low cost, chemical inertness, and low permeability. Here crystallinity of the linear material plays a key role.

∗Collaboratory for Advanced Computing and Simulations, University of Southern California, Los Angeles, CA, U.S.

For example, a high-pressure hydrogen permeation test has shown HDPE is 2.5 times less permeable than the low density polyethylene (LDPE) that has low crystallinity and amorphous regions [Fujiwara et al., 2021].

Polyethylene (PE) consists of chains of CH\textsubscript{2} repeat unit and has been extensively studied for many scientific and engineering applications. Figure 1(a) and (b) present crystalline PE and amorphous PE systems, respectively. The tortuosity is defined as the ratio between theoretical straight path over the actual diffusion trajectory length, which is an important factor to understand the permeability. Figure 1(c) schematically presents H\textsubscript{2} diffusion pathway in polymer linear, in which the blue blocks represent highly crystalline regions with little permeability while the other area is filled by amorphous phases thus more permeable.

Molecular Dynamics (MD) simulation is a powerful computational tool to study solubility and permeability in polymers at the atomic level [Kotelyanskii and Theodorou, 2004]. However, the accessible timescale using MD is severely limited due to the computational cost, thus impractical to study diffusion phenomena that takes over the order of milliseconds. Reinforcement learning (RL) is a promising approach to discover energy efficient diffusion pathway in the complex energy landscape, akin to the maze-solving problem. RL has been used to study protein structure prediction [Soltanikazemi et al., 2022, Yang et al., 2022] and drug design [Korshunova et al., 2022, Atance et al., 2022]. Another advantage of RL is the ability to incorporate dynamically varying environments such as the low-energy conformation changes in polymer chains [Padakandla, 2021].

2 Method

2.1 Reinforcement Learning

In Reinforcement learning (RL), an agent interacts with environment to learn optimal policy from their actions and received rewards [Sutton and Barto, 2018].

Q-learning [Watkins, 1989] is a value-based learning to find optimal policy to maximize the cumulative reward. The optimal action-value function (Eq. 1) is iteratively updated given action \( A \) in state \( S \) at step \( t \), and \( R_{t+1} \) is the reward for the next action (Eq. 2). Here, \( \alpha \) is the learning rate and \( \gamma \) is discount factor that controls the extent of future rewards for an agent to take into account.

\[
Q^*(s, a) = \mathbb{E}[r + \gamma \max_{a'} Q(s', a') | s, a]
\]  

\[
Q^*(S_t, A_t) \leftarrow Q^*(S_t, A_t) + \alpha[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)]
\]

2.2 Deep Q-learning

Deep reinforcement learning (DRL) uses deep neural networks combined with RL to handle complex decision-making tasks [Arulkumaran et al., 2017, Li, 2017]. DRL has been used in numerous applications such as autonomous control [Zhu et al., 2016], game playing [Mnih et al., 2015], and natural language processing [Bahdanau et al., 2016]. Deep Q network (DQN) introduces Convolutional Neural Networks (CNN) to approximate the optimal Q value, i.e. \( Q^*(s, a; \theta) \approx Q^*(s, a) \) where \( \theta \) is
network parameter, so that the complex state of Atari games can be incorporated. The parameter \( \theta \) is trained by minimizing the loss function \( L(\theta) \) as below,

\[
L(\theta) = \mathbb{E}_{(s, a, r, s') \sim D}[(r + \gamma \max_{a'} Q(s', a'; \theta) - Q(s, a; \theta))^2],
\]

where \( D \) is the experience replay buffer that stores a tuple of agent’s experience, \( e_t = (s_t, a_t, r_t, s_{t+1}) \) at each time-step. During the training, the minibatch size of experiences are sampled from the replay buffer. \( \epsilon \)-greedy policy is applied to enhance the agent’s exploration while exploring knowledge from previous training. To reduce overfitting and the learning more stable, the target network is periodically cloned from behavioral network.

Many algorithmic extensions to the original DQN have been proposed to date including Double Q-learning [van Hasselt et al., 2015], prioritized replay buffer [Schaul et al., 2015; Fedus et al., 2020], Dueling networks [Wang et al., 2015], Multi-step learning [Sutton, 1988], Distributional RL [Bellemare et al., 2017], Noisy networks [Fortunato et al., 2017]. These extensions are collectively called Rainbow DQN [Hessel et al., 2017] and also utilized in our framework.

Next, we describe each element of our framework.

**Environment:** Environment is modeled by reactive MD (RMD) simulation using a reactive inter-atomic potential, ReaxFF [Senftle et al., 2016]. ReaxFF employs the bond-order concept and the dynamical charge scheme called QEq and accurately describes the interatomic interactions for hydrocarbon and polymeric systems [Duin et al., 2001; Vashisth et al., 2018]. All RMD simulations were carried out using a scalable MD software RXMD [Nomura et al., 2020]. Pytorch 1.12.0+cu102 [Paszke et al., 2019] and Ray 2.2.0 [Moritz et al., 2018] are used for model training and the interprocess communication, respectively.

**Agent:** An agent is modeled by a harmonic potential \( \frac{1}{2} k_s (\vec{r} - \vec{r}_0)^2 \) where \( k_s \) is the spring constant, \( \vec{r} \) is the coordinates of an atom bound to the agent, and \( \vec{r}_0 \) is the agent’s position. See Fig.2(a). The agent is initially placed near the \( y-z \) plane at \( x = 0 \). When the agent makes an action, one of the five displacement vectors \( a = \{(1, 0, 0), (0, 1, 0), (0, -1, 0), (0, 0, 1), (0, 0, -1)\} \) is selected to update the position of agent. Such a discrete action may suffer from the action oscillation problem [Chen et al., 2021], thus we mask the displacement vector \((−1, 0, 0)\) in this study. After each action, the system is briefly relaxed to sample the potential energy at the new state.

**State:** The state is a three-dimensional grid that represents the local atomic density around the agent’s location. See Fig.2(b). Within a cutoff distance of 5 Å, we use the Gaussian Kernel to compute the density contribution from each neighbor atom.

**Reward:** The reward consists of five functions: \( R_{\text{position}}, R_{\text{energy}}, R_{\text{density}}, R_{\text{distance}}, \) and \( R_{\text{time}} \). \( R_{\text{position}} \) is a monotonically increasing function based on the \( x \)-coordinate of the agent. \( R_{\text{energy}} \) encourages to find a lower energy state than the past history, \( R_{\text{density}} \) keeps the agent from colliding with neighbor atoms. \( R_{\text{distance}} \) keeps the agent and a hydrogen atom together. We also apply a time penalty to avoid agent staying at the same location for long time. In addition, the agent receives an end-of-episode reward when it reaches the goal, i.e. within 2 Å from the right end of the simulation box.

Figure 2: (a) Agent modeled by harmonic potential, (b) state, and (c) Q-function using CNN and fully-connected layers. The network takes the local atomic density distribution at a given state to infer the Q-values.
Figure 3: Agent’s performance using crystalline PE system. (a) Agent’s $x$-coordinate at the end of episode and (b) total reward as a function of the number of RL agents $N = 1, 4, \text{ and } 16$ respectively.

**Q-function:** Three-dimensional CNN is used to model the Q-function that consists of three convolutional (conv) layers with ReLu activation function [Agarap, 2018], followed by two fully-connected (FC) layers (Fig. 2 (c)). For each conv layer, kernel size and strides are $(8, 2), (4, 1)$ and $(3, 1)$ respectively. We use 32, 64, and 128 channels for each conv layer. 512 and 80 nodes are used for the first and second FC layers.

3 Experiment

We have tested our framework on crystal and amorphous PE systems. To generate the crystalline system, we replicate the unit cell of PE ([Bunn, 1939]) $4 \times 5 \times 11$ times in each direction. The periodic boundary condition is applied on all three directions. The obtained system dimensions are $29.6 \times 24.65 \times 27.87$ (Å$^3$) that contains 2,640 atoms in total with the density of roughly 1 g/cc. For the amorphous system, we first generate single PE chain consists of 50 atoms. Packmol package [Martínez et al., 2009] is used to create a supercell with the dimensions of $(32 \ \text{Å})^3$. Forty PE chains are placed in the system with a tolerance of $2.0 \ \text{Å}$ separation between the chains. We thermalize the system at room temperature while gently compress the system with a constant compression ratio. The final system size $(24.86 \ \text{Å})^3$ for the amorphous PE system. Total number of atoms is 2,000 at the density around 0.97 g/cc.

Figure 3 (a) and (b) present the agent’s final $x$-coordinate and the total reward as a function of the number of agents $N$. Overall, the agent has found a diffusion path to reach the goal at $x = 27 \ \text{Å}$. See Fig 3 (a). The final reward quickly increases with $N = 16$ and reaches to a steady value around 120 after 10,000 steps. With $N = 4$, the final reward has become a similar value as the $N = 16$ case. On the other hand, it saturates around 80 with $N = 1$ indicating the agent being trapped by a sub-optimal diffusion path. See Fig 3 (b).

After obtaining the energy barriers along diffusion pathway, we estimate the diffusion time $T_m$ based on transition state theory as $T_m = \sum_i \frac{\hbar}{k_B T} \exp\left(\frac{E^{(i)}_A}{k_B T}\right)$, where $E^{(i)}_A$ is the $i$-th energy barrier along an energy profile, which is obtained by the difference between an energy minimum and subsequent energy maximum. $\hbar$ is the reduced Planck constant, $k_B$ is Boltzmann constant, $T$ is the temperature and set to be at 300 K. The speed of molecular diffusion is obtained from the size of simulation system (29.6 Å for the crystalline and 32 Å for the amorphous systems) divided by the total diffusion time.

In the crystal system using $N = 16$ agents, we obtained the diffusion speed of 0.589 nm/day. While the agent successfully finished episode with the $N = 1$, it failed to find an energy efficient pathway resulting in an infinite $T_m$. Table 1 summarizes the best $T_m$ for both crystalline and amorphous systems. First of all, $T_m$ in the amorphous system is greater than the ones in the crystal system, which suggests that the agent has correctly learned the energy landscape difference between the two systems. Overall trend in $T_m$ agrees with the agent’s performance, however, it is also very sensitive to the energy barriers $E_A$, which can be influenced by a slight fluctuation in the diffusion pathways. Currently we are developing piecewise parallel Nudged Elastic Band [Henkelman et al., 2000] to refine the obtained energy barriers with robust diffusion time estimate.
Table 1: $\text{H}_2$ diffusion speed (nm/day) in crystal and amorphous PE systems.

<table>
<thead>
<tr>
<th>Number of Agents N</th>
<th>1</th>
<th>4</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crystal</td>
<td>N/A</td>
<td>2.25 $\times$ 10^{-5}</td>
<td>0.589</td>
</tr>
<tr>
<td>Amorphous</td>
<td>41.35</td>
<td>521.052</td>
<td>1,846.42</td>
</tr>
</tbody>
</table>

4 Conclusions

We have developed a DRL framework to study molecular diffusion through polymeric materials. Using the efficient model training based on the distributed replay buffer, an ensemble of RL agents quickly learns the complex energy landscape of the system to uncover energy efficient pathways. Subsequently, the diffusion time of each pathway is estimated using transition state theory. The distributed training with 16 agents shows a significant improvement in the training metrics as well as the diffusion time.

Broader Impact

The RL framework presented in this study is system-agnostic and easily applicable to many molecular diffusion processes. It does not require domain expert knowledge nor prescribed reaction coordinates to learn and uncover energy-efficient pathways. With the capability to access experimentally relevant timescale using transition state theory without sacrificing atomistic level insights, our framework has a potential to find many applications in engineering and scientific problems.

Acknowledgments

Research was supported by the U.S. Department of Energy, Office of Basic Energy Sciences, Division of Materials Sciences and Engineering, Neutron Scattering and Instrumentation Sciences program under Award DE-SC0023146. This work was supported by the Ershaghi Center for Energy Transition (E-CET).

References


