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) [Fujiwara 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.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

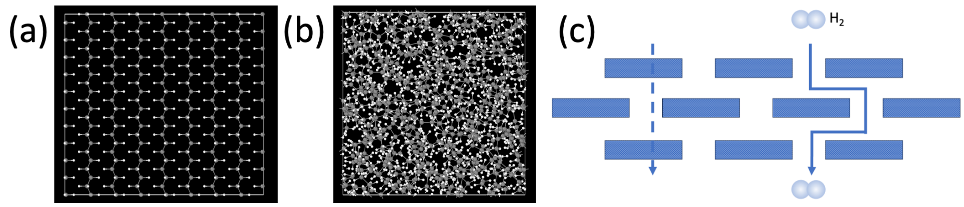

Figure 1: (a) and (b) crystalline and amorphous polyethylene systems. White and gray spheres represent hydrogen and carbon atoms. (c) molecular diffusion through crystalline (blue) and amorphous (white) regions. Theoretical straight path (dotted line) and actual diffusion path (solid line).

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 region[Fujiwara et al., 2021].

Polyethylene (PE) consists of chains of CH_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. Fig.1 (c) schematically presents H_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, α is the learning rate and γ is discount factor that controls the extent of future rewards for an agent to take into account.

$$Q^*(s,a) = \mathbb{E}[r + \gamma \max_{a'} Q(s',a')|s,a] \tag{1}$$

$$Q^*(S_t,A_t) \leftarrow Q^*(S_t,A_t) + \alpha[R_{t+1} + \gamma \max_{a} Q(S_{t+1},a) - Q(S_t,A_t)] \tag{2}$$

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 θ is

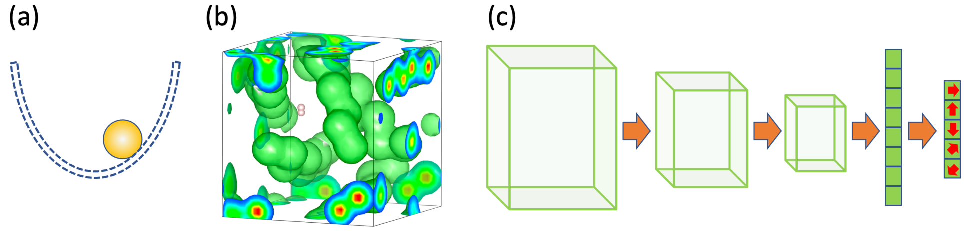

Figure 2: (a) Agent modeled by harmonic potential, (b) state, and (c) Q-function using CNN and fully-connected layers. The network takes the local atomic density distribution at a given state to infer the Q-values.

network parameter, so that the complex state of Atari games can be incorporated[Mnih et al., 2013]. The parameter θ is trained by minimizing the loss function $L(\theta)$ as below,

$$L(\theta) = \mathbb{E}_{(s,a,r,s')\sim D}[(r + \gamma \max_{a'} Q(s', a'; \theta^-) - Q(s, a; \theta))^2], \tag{3}$$

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. ϵ-greedy policy is applied to enhance the agent's exploration while exploring knowledge from previous training. To reduce overefitting 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 interatomic 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, repsectively.

Agent: An agent is modeled by a harmonic potential $1/2k_s(\vec{r} - \vec{r_0})^2$ 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: $R_{position}$, R_{energy}, $R_{density}$, $R_{distance}$, and R_{time}. $R_{position}$ is a monotonically increasing function based on the x-coordinate of the agent. R_{energy} encourages to find a lower energy state than the past history, $R_{density}$ keeps the agent from colliding with neighbor atoms. $R_{distance}$ 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.

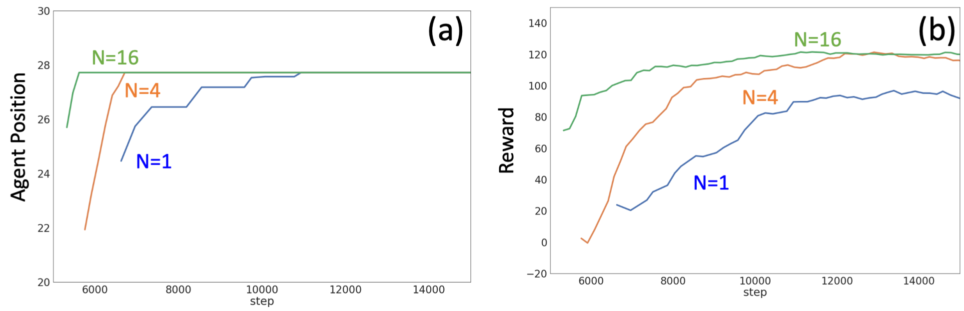

Figure 3: Agent's performance using crystalline PE system. (a) Agent's x-coordinate at the end of episode and (b) total reward as a function of the number of RL agents N = 1, 4, and 16 respectively.

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 1g/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 Å)^3. Forty PE chains are placed in the system with a tolerance of 2.0 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 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 $x = 27$ 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 $T_m = \sum_i \frac{\hbar}{k_B T} * exp(\frac{E_A^{(i)}}{k_B T})$, where $E_A^{(i)}$ 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 Plank 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: H_2 diffusion speed (nm/day) in crystal and amorphous PE systems.

Number of Agents N	1	4	16
Crystal	N/A	2.25×10^{-5}	0.589
Amorphous	41.35	521.052	1,846.42

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

Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.

Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34(6):26–38, August 2017.

Sara Romeo Atance, Juan Viguera Diez, Ola Engkvist, Simon Olsson, and Rocío Mercado. De novo drug design using reinforcement learning with graph-based deep generative models. *Journal of Chemical Information and Modeling*, 62:4863–4872, October 2022.

Dzmitry Bahdanau, Philemon Brakel, Kelvin Xu, Anirudh Goyal, Ryan Lowe, Joelle Pineau, Aaron C. Courville, and Yoshua Bengio. An actor-critic algorithm for sequence prediction. *arXiv preprint arXiv:1607.07086*, 2016.

Marc G. Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. *arXiv preprint arXiv:1707.06887*, July 2017.

C. W. Bunn. The crystal structure of long-chain normal paraffin hydrocarbons. the "shape" of the ch2 group. *Trans. Faraday Soc.*, 35:482–491, 1939.

Chen Chen, Hongyao Tang, Jianye Hao, Wulong Liu, and Zhaopeng Meng. Addressing action oscillations through learning policy inertia, 2021.

Adri C.T. Van Duin, Siddharth Dasgupta, Francois Lorant, and William A. Goddard. Reaxff: A reactive force field for hydrocarbons. *Journal of Physical Chemistry A*, 105:9396–9409, October 2001.

William Fedus, Prajit Ramachandran, Rishabh Agarwal, Yoshua Bengio, Hugo Larochelle, Mark Rowland, and Will Dabney. Revisiting fundamentals of experience replay. *arXiv preprint arXiv:2007.06700*, July 2020.

Meire Fortunato, Mohammad Gheshlaghi Azar, Bilal Piot, Jacob Menick, Ian Osband, Alex Graves, Vlad Mnih, Remi Munos, Demis Hassabis, Olivier Pietquin, Charles Blundell, and Shane Legg. Noisy networks for exploration. *arXiv preprint arXiv:1706.10295*, June 2017.

Hirotada Fujiwara, Hiroaki Ono, Keiko Ohyama, Masahiro Kasai, Fumitoshi Kaneko, and Shin Nishimura. Hydrogen permeation under high pressure conditions and the destruction of exposed polyethylene-property of polymeric materials for high-pressure hydrogen devices (2)-. *International Journal of Hydrogen Energy*, 46: 11832–11848, March 2021.

Graeme Henkelman, Blas P. Uberuaga, and Hannes Jónsson. Climbing image nudged elastic band method for finding saddle points and minimum energy paths. *Journal of Chemical Physics*, 113:9901–9904, 12 2000.

Matteo Hessel, Joseph Modayil, Hado van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. *arXiv preprint arXiv:1710.02298*, October 2017.

Maria Korshunova, Niles Huang, Stephen Capuzzi, Dmytro S. Radchenko, Olena Savych, Yuriy S. Moroz, Carrow I. Wells, Timothy M. Willson, Alexander Tropsha, and Olexandr Isayev. Generative and reinforcement learning approaches for the automated de novo design of bioactive compounds. *Communications Chemistry*, 5, December 2022.

Michael Kotelyanskii and Doros N. Theodorou. *Simulation Methods for Polymers*. Marcel Dekker, 2004.

Yuxi Li. Deep reinforcement learning: An overview. *arXiv preprint arXiv:1701.07274*, January 2017.

Leandro Martínez, Ricardo Andrade, Ernesto G Birgin, and José Mario Martínez. Packmol: A package for building initial configurations for molecular dynamics simulations. *Journal of Computational Chemistry*, 30: 2157–2164, 2009.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, December 2013.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518:529–533, February 2015.

Philipp Moritz, Robert Nishihara, Stephanie Wang, Alexey Tumanov, Richard Liaw, Eric Liang, Melih Elibol, Zongheng Yang, William Paul, Michael I. Jordan, and Ion Stoica. Ray: A distributed framework for emerging ai applications. *arXiv preprint arXiv:1712.05889*, 2018.

Ken-ichi Nomura, Rajiv K. Kalia, Aiichiro Nakano, Pankaj Rajak, and Priya Vashishta. Rxmd: A scalable reactive molecular dynamics simulator for optimized time-to-solution. *SoftwareX*, 11, January 2020.

Sindhu Padakandla. A survey of reinforcement learning algorithms for dynamically varying environments. *ACM Computing Surveys*, 54(6):1–25, jul 2021. doi: 10.1145/3459991. URL `https://doi.org/10.1145%2F3459991`.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, November 2015.

Thomas P. Senftle, Sungwook Hong, Md Mahbubul Islam, Sudhir B. Kylasa, Yuanxia Zheng, Yun Kyung Shin, Chad Junkermeier, Roman Engel-Herbert, Michael J. Janik, Hasan Metin Aktulga, Toon Verstraelen, Ananth Grama, and Adri C.T. Van Duin. The reaxff reactive force-field: Development, applications and future directions, March 2016.

Elham Soltanikazemi, Raj S. Roy, Farhan Quadir, and Jianlin Cheng. A deep reinforcement learning approach to reconstructing quaternary structures of protein dimers through self-learning. *BioRxiv (Cold Spring Harbor Laboratory)*, April 2022.

Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine Learning*, 1988.

Richard S Sutton and Andrew Barto. *Reinforcement Learning : an Introduction*. The Mit Press, 2018.

Muhammad R. Usman. Hydrogen storage methods: Review and current status. *Renewable and Sustainable Energy Reviews*, 167, October 2022.

Hado van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. *arXiv preprint arXiv:1509.06461*, September 2015.

Aniruddh Vashisth, Chowdhury Ashraf, Weiwei Zhang, Charles E. Bakis, and Adri C.T. Van Duin. Accelerated reaxff simulations for describing the reactive cross-linking of polymers. *Journal of Physical Chemistry A*, 122:6633–6642, August 2018.

Ziyu Wang, Tom Schaul, Matteo Hessel, Hado van Hasselt, Marc Lanctot, and Nando de Freitas. Dueling network architectures for deep reinforcement learning. *arXiv preprint arXiv:1511.06581*, November 2015.

Christopher John Cornish Hellaby Watkins. *Learning from Delayed Rewards*. PhD thesis, King's College, Cambridge, UK, May 1989.

Kaiyuan Yang, Houjing Huang, Olafs Vandans, Adithya Murali, Fujia Tian, Roland H.C. Yap, and Liang Dai. Applying deep reinforcement learning to the HP model for protein structure prediction. *Physica A: Statistical Mechanics and its Applications*, 609:128395, November 2022.

Thomas A. Yersak, Daniel R. Baker, Yuka Yanagisawa, Stefan Slavik, Rainer Immel, André Mack-Gardner, Michael Herrmann, and Mei Cai. Predictive model for depressurization-induced blistering of type iv tank liners for hydrogen storage. *International Journal of Hydrogen Energy*, 42:28910–28917, November 2017.

Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J. Lim, Abhinav Gupta, Li Fei-Fei, and Ali Farhadi. Target-driven visual navigation in indoor scenes using deep reinforcement learning. *arXiv preprint arXiv:1609.05143*, September 2016.