

# CryoRL: Reinforcement Learning Enables Efficient Cryo-EM Data Collection

Quanfu Fan  
 Amazon  
 quanfu@amazon.com

Yilai Li  
 University of Michigan  
 yilai@umich.edu

Yuguang Yao  
 Michigan State University  
 yaoyugua@msu.edu

John Cohn  
 IBM  
 johncohn@us.ibm.com

Sijia Liu  
 Michigan State University  
 liusiji5@msu.edu

Ziping Xu  
 Harvard University  
 zipingxu@fas.harvard.edu

Seychelle Vos  
 MIT  
 seyvos@mit.edu

Michael Cianfrocco  
 University of Michigan  
 mcianfro@umich.edu

## Abstract

*Single-particle cryo-electron microscopy (cryo-EM) has become one of the mainstream structural biology techniques because of its ability to determine high-resolution structures of dynamic bio-molecules. However, cryo-EM data acquisition remains expensive and labor-intensive, requiring substantial expertise. Structural biologists need a more efficient and objective method to collect the best data in a limited time frame. We formulate the cryo-EM data collection task as an optimization problem in this work. The goal is to maximize the total number of good images taken within a specified period. We show that reinforcement learning offers an effective way to plan cryo-EM data collection, successfully navigating heterogenous cryo-EM grids. The approach we developed, cryoRL, demonstrates better performance than average users for data collection under similar settings.*

## 1. Introduction

Single-particle cryo-electron microscopy (cryo-EM) has become one of the mainstream structural biology techniques due to its ability to solve the structures of many bio-molecules with moderate heterogeneity and without the need for crystallization. In recent years, continued software development has led to automation in both data collection and image processing [1]. Moreover, with the improvement of the detectors and microscopes techniques, data acquisition has been dramatically accelerated [6, 34].

Cryo-EM serves as a critical tool in the development of vaccines and therapeutics to combat COVID-19 by SARS-CoV-2. Within weeks of the release of the genomic se-

quence of SARS-CoV-2, cryo-EM determined the first SARS-CoV-2 spike protein structure [37]. Since this original publication, cryo-EM was used to determine additional SARS-CoV-2 structures such as spike protein bound to antibody fragments [20, 27], remdesivir bound to SARS-CoV-2 RNA-dependent RNA polymerase [5, 13, 40], and reconstructions of intact SARS-CoV-2 virions [11, 38].

Despite these advances, cryo-EM data collection remains *ad-hoc*, rudimentary, and subjective. Due to variations in sample quality across a cryo-EM grid, users collect images at different magnifications ranging from resolutions of 0.66 nm to 500 Å. Significant user expertise enables experts to define and refine locations suitable for data collection. To provide objective feedback, "on-the-fly" image processing [16, 31] can confirm high-quality regions on the cryo-EM sample. Despite this information, data collection remains highly subjective. Cryo-EM is also an expensive technique, further compounding challenges faced by users. Purchasing, preparing, and installing a top cryo-electron microscope can cost about \$10 to 20 million USD, and the daily operational cost can be around \$10,000 USD [9]. Therefore, structural biologists need methods that can help collect the best data possible in a limited amount of time.

In this paper, we formulate the data collection problem as an optimization task, where the goal is to learn intelligent strategies from data to guide the microscope movement, possibly via manual suggestion or robotic manipulation. We model the optimization problem as a Markov decision process and propose to solve it by combining supervised classification and deep reinforcement learning (RL) [30]. We present a new data acquisition algorithm, cryoRL, which enables data collection with no subjective decisions, no user intervention, and increased efficiency. To address the poten-

tial enormously large action space in our problem, we further propose to eliminate irrelevant or sub-optimal actions based on the classification results to enable effective policy exploration, which improves cryoRL in both efficiency and accuracy. As compared with human subjects, cryoRL achieves better performance than average users. To the best of our knowledge, cryoRL is the first AI-based algorithm in cryo-EM data acquisition such that a policy is learned and can directly help the user steer the microscope.

We collected datasets on different grid types to design, implement, and test cryoRL. The first of its kind, our data collection involves no user decision; instead, we selected the areas from a systematic pattern of data collection to obtain images of all holes and micrographs (See Section 3 for details of our data). Our datasets will be released to the public to serve as a critical benchmark for evaluating cryo-EM data collection algorithms.

Our high-level conclusions and contributions include:

- CryoRL achieves better performance than other popular optimization techniques such as Genetic Algorithm [35] and Simulated Annealing [12], and demonstrates good generalization capability;
- CryoRL with our proposed invalid action elimination runs 2~3 times faster than the vanilla DQN baseline while enabling more robust policy learning;
- CryoRL offers a new approach to cryo-EM data collection that demonstrates promising results by outperforming average users in a human performance study;
- We provide a first-kind-of cryo-EM dataset that is critical for algorithm development and benchmarking.

## 2. Related Work

There are currently no automated, 'intelligent' cryo-EM data collection approaches. Instead, subjective decision-making drives cryo-EM data acquisition. To guide user-driven data collection, on-the-fly image analysis provides feedback on data quality, including Appion [16], Warp [31], and cryoSPARC Live. To provide more objective measures of data quality to users, researchers have developed a pre-trained deep learning-based micrograph assessment model [21] and downstream on-the-fly data processing [29]. However, despite these efforts, on-the-fly processing requires a sizeable number of micrographs before providing useful feedback. Data collection requires user training to develop expertise to guide data collection in the most efficient manner possible.

Reinforcement learning (RL) has been widely applied to address practical optimization problems such as network planning [44], vehicle routing [23], on-line recommendation [43], robot trajectory optimization [14] as well as game playing [28]. Some practical applications have adopted RL for enabling fast data collection or processing. For example in [32, 42], RL-based methods are proposed to optimize

unmanned aerial vehicle flight trajectories for efficient data collection. In [15], a deep RL agent is used to condition the state of the probe for autonomous Scanning Probe Microscopy (SPM). Nevertheless, automating data collection by machine learning techniques in real-world scenarios remains an under-explored problem.

Large action spaces are a common problem to deal with in RL. Existing techniques include action masking [3, 39] to mask out invalid actions, action elimination [41] to remove inferior actions, and action reshaping [10] to transform a discrete action space to a simpler one or a continuous one. Our proposed action elimination is in a similar spirit to Action Elimination Network (AEN) [41], but instead relies on the estimated quality of a hole to exclude invalid actions directly rather than learning to reduce the action space.

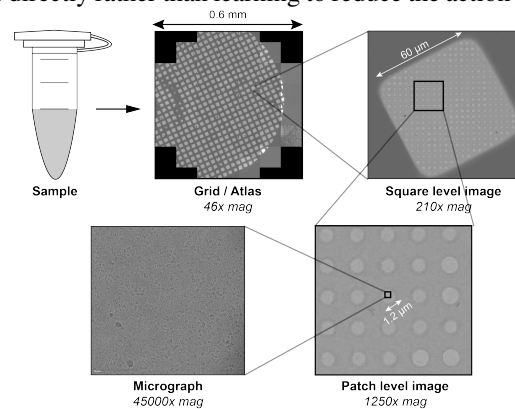


Figure 1. Overview of cryo-EM data collection. A purified sample is prepared and vitrified on the support grid. The atlas image provides a low magnification overview by stitching multiple "grid-level" images into a single montage. Next, users will select specific squares to image at medium magnification. After inspection, the user selects "patch" areas on the square to inspect holes with higher magnification, using the patch image to decide holes to collect for micrographs. The micrographs contain high-resolution images for downstream data processing.

## 3. Cryo-EM Data Collection

The general practice of data acquisition in cryo-EM is abstracted in Fig. 1. Typically, a purified biological sample is dispensed and vitrified onto a grid comprised of gold or copper support bars. A grid contains a mesh of squares, and each square has a lattice of regularly-spaced holes. Ideally, within each hole, there are vitrified single-particles related to the sample of interest, where data collection amounts to users recording images of each hole as micrographs.

Cryo-EM samples exhibit heterogeneity across the specimen. Whereas there are many local correlations between squares and holes on the grid, many holes are empty, contain aggregates, or contain non-vitreous ice contamination. The user has no prior knowledge of such distribution until the square-level or hole-level images are acquired, which

can be captured by the microscope by changing to different magnifications. Note that each greater magnification requires significant time for the microscope to move and settle. Moreover, because the time on the microscope is precious and limited, data collection can typically cover less than 1% of the total grid. The user needs to navigate through the "grid-square-hole" hierarchy and collect the best micrographs in a limited time.

In this paper, we suppose that a user preselects a set of squares and patch-level images by a quick atlas survey. We formulate the data acquisition task to find the highest quality holes and plan the overall data collection route with cryo-RL. Although this is not the traditional way people collect cryo-EM data, we believe such prerequisites provide a global understanding of the hole quality distribution on a grid by taking a series of low to medium magnification square and patch level images.

Each micrograph has an objective measure of data quality, which is the goodness-of-fit for the frequency domain when estimating the defocus of the micrograph. We introduce the term "CTFMaxRes" to be the maximum resolution (Å) for the fit of the contrast transfer function (CTF) to a given micrograph using the program CTTFIND4 [25]. CTFMaxRes is calculated from the 1D power spectrum of the micrograph and estimates the maximum resolution for the detected CTF oscillations [4]. The field of cryo-EM utilizes CTFMaxRes to provide an indirect metric for data quality. In general, the lower this value, the higher the quality of the micrograph. CryoRL will predict the quality of each hole from the patch-level image using an image classifier (Section 4). For simplicity, we define CTFMaxRes as the CTF value for this paper.

## 4. RL-based Approach

### 4.1. Overview

During data collection, a user needs to make decisions based on the quality of images taken at different magnification levels: grid, square, and patch-level. Given that the data are visually similar and there are significant costs (time) of moving to other grid areas and refocusing, there is no easy planning that a user can make manually in a regular data collection. As a result, the user explores only a small portion of a grid, making the data collection process inefficient and subjective.

In this work, we formulate the data collection problem as planning an optimal path for operating the microscope. The goal is to move the microscope to explore desired places on a grid in a given amount of time, with the operational cost constraints taken into account (Section 4.2). We propose to solve the path planning problem by RL, a technique that has demonstrated success in many vision applications [18]. Compared to other widely used optimization solvers such

as Genetic Algorithm (GA) [35] and Simulated Annealing (SA) [12], RL is more suitable for modeling sequential problems and possibly less heuristic in system design.

As illustrated in Fig. 2, our proposed approach combines an image classifier and an RL network to enable automatic planning of microscope movement. The supervised classifier categorizes a hole into low or high quality based on its CTF value. Efficient hierarchical feature representations for cryo-EM images at different magnification levels are generated from the classification results. These features, along with the observation history, are exploited to train a deep Q-network (DQN) [22] to assess the status of all the unvisited holes and suggest the best holes to look at next. We further design a rewarding mechanism to drive the learning of DQN. The design in general values small microscope movements to avoid wasted time. For example, moving to a different patch on the same grid-level image receives a higher score than changing to an entirely new grid-level image (Section 4.3). Finally, to handle the potentially large action space in our problem, we propose a method to eliminate invalid actions, which not only results in a significant speedup of CryoRL by 2~3 times, but also improves the robustness of the approach (Section 4.3).

As mentioned earlier, a human user can usually cover a small portion of the grid during a data collection session. In contrast, one significant advantage of our proposed approach is that it allows for a substantially larger exploration of the grid by the microscope as the approach learns to focus on promising regions with high-quality data. We demonstrate in Section 5 that our system is highly effective, achieving comparable performance to human subjects.

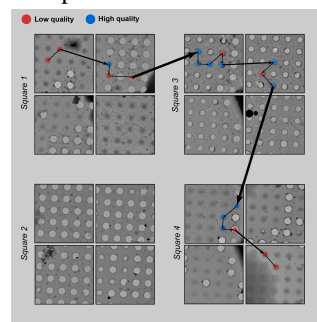


Figure 3. A schematic illustration of a path showing the microscope movement planned in a data collection session. Different microscopic operations are associated with different costs, which are indicated by the edge width.

### 4.2. Problem Formulation

As previously described, cryo-EM data collection is steering the microscope hierarchically at different magnification levels to explore a grid to identify high-quality micrographs. This sequential process involves several mechanical operations to allow microscope navigation to dif-

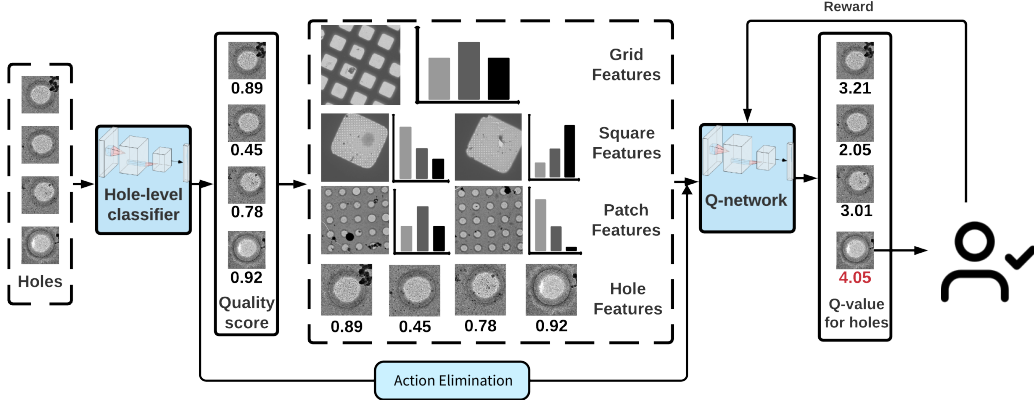


Figure 2. cryoRL-guided cryo-EM data collection. cryoRL consists of an offline hole-level classifier to estimate the quality of holes and a deep Q-network to learn effective strategies for steering the microscope for data collection. The hole classifier outputs serve as features of the RL network. The agent (or user) provides feedback (rewards) to the system according to the microscope movement and the measured quality of the micrographs taken for the holes recommended by the system. In addition, to address the potential issue of large action spaces in our problem, we developed an efficient method to eliminate invalid actions based on the quality of holes.

ferent regions of a grid. The process of data collection involves area switching (changing to a new grid-level image), square switching (changing to a new square-level image), and patch switching (changing to a new patch-level image). Since the data distribution is non-uniform on a grid and it takes time to prepare the microscope for imaging at different levels, an automatic method to guide the data exploration more intelligently will improve data quality and efficiency for data collection.

As shown in Fig. 3, an effective data collection session aims at finding a sequence of holes where there is a considerable portion of high-quality micrographs. Let  $\mathcal{H} = \{h_l \mid l = 1 \dots n_h\}$  be a sequence of holes in a set of patches  $\mathcal{P}$  sampled from different square-level and grid-level images ( $\mathcal{S}$  and  $\mathcal{G}$ ) by the user. We denote  $\mathcal{P}_{h_l}$ ,  $\mathcal{S}_{h_l}$  and  $\mathcal{G}_{h_l}$  as the corresponding patch-level, square-level and grid-level images of  $h_l$ , respectively. Also,  $\text{ctf}(h_l)$  is a function representing the CTF value of a hole  $h_l$ . Our goal is to identify a maximum subset of holes from  $\mathcal{H}$  with low-CTF values in a given amount of time  $\tau$ . Mathematically, this is equivalent to optimizing an object function as follows,

$$\max \sum_{l=0}^{n_h-1} \rho(h_l) - c(t(h_l, h_{l-1})) \quad \text{s.t.} \quad \sum_{l=0}^{n_h-1} t(h_l) \leq \tau \quad (1)$$

where  $\rho(h_l)$  be such an indicator function for a hole  $h$  that

$$\rho(h_l) = \begin{cases} 1 & \text{if } \text{ctf}(h_l) \leq 6.0 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

and  $c: \mathbb{R} \mapsto \mathbb{R}$  is a cost function associated with the corresponding microscope operation and determined by the total amount of time  $t(h_l, h_{l-1})$  the time it takes to move from

$h_{l-1}$  to  $h_l$ , i.e.  $t(h_l, h_{l-1})$  is given by

$$\begin{cases} 2.0 & \text{if } \mathcal{P}_{h_{l-1}} = \mathcal{P}_{h_l} \text{ (same patch)} \\ 3.0 & \text{if } \mathcal{P}_{h_{l-1}} \neq \mathcal{P}_{h_l}, \mathcal{S}_{h_{l-1}} = \mathcal{S}_{h_l} \text{ (same square)} \\ 5.0 & \text{if } \mathcal{S}_{h_{l-1}} \neq \mathcal{S}_{h_l}, \mathcal{G}_{h_{l-1}} = \mathcal{G}_{h_l} \text{ (same grid)} \\ 10.0 & \text{if } \mathcal{G}_{h_{l-1}} \neq \mathcal{G}_{h_l} \text{ (different grid)} \end{cases}$$

Note that the time  $t$  above is set in a way so that it highly corresponds to the natural time of the microscope movement in real-world scenarios. Nevertheless, in practice, it can be more precisely calculated based on the distance of the microscope movement and other factors.

By setting  $r(h_l) = \rho(h_l) - c(t(h_l, h_{l-1}))$ , we can further rewrite Eq. 1 as

$$\max \sum_{l=0}^{n_h-1} r(h_l) \quad \text{s.t.} \quad \sum_{l=0}^{n_h-1} t(h_l, h_{l-1}) \leq \tau \quad (3)$$

Eq. 3 has the same form as the standard accumulative reward (without a discount factor) that is maximized in RL [30]. In what follows, we describe how to design a RL system to solve the path optimization problem in Eq. 3.

### 4.3. Path Optimization by Reinforcement Learning

We study the cryo-EM data acquisition task by RL, where an agent interacts with environment (i.e. the grid here) by sequentially selecting holes for taking micrographs over a sequence of time steps, with an objective to maximize the cumulative reward described in Eq. 3. We briefly describe the basic components of our system as follows.

*Environment:* the atlas or grid.

*Agent:* a robot or user steering the microscope.

*States.* Let  $u_i \in \{0, 1\}$  be a binary variable denoting the status of hole, i.e. visited or unvisited. Then

a state  $s$  in our setting can be represented by a sequence of holes and their corresponding statuses  $s = \langle (h_1, u_1), (h_2, u_2), \dots, (h_n, u_n) \rangle$ , where  $n$  is the total number of holes.

**Actions.** An action  $a_i$  of the agent in our system is to move the microscope to the next target hole  $h_i$  for imaging. Note that in our case, any unvisited hole has a chance to be picked by the agent as a target, thus the action space is large. Also, during tests, the number of holes (i.e actions) is unknown. Instead of adopting more sophisticated methods to handle continuous action space as proposed in [17, 19], we simply modify the Q-network to estimate the Q-value for every single hole rather than all of them at once. We show this suffices for handling the large action space in our case.

**Rewards.** We assign a positive reward 1.0 to the agent if an action results in a target hole with a CTF value less than  $6.0\text{\AA}$  and 0.0 otherwise. The agent also receives a negative reward depending on the operational cost associated with a hole visit. Specifically, we model the negative reward as  $c(t) = 1.0 - e^{-\beta(t-t_0)}$  ( $\beta > 0, t_0 \geq 0$ ). We empirically set  $\beta$  and  $t_0$  to 0.185 and 2.0, which define the final reward function for our RL system as,  $r(a_i, h_l) =$

$$\begin{cases} 1.0 & \text{if } \text{ctf}(h_i) < 6.0 \ \& \ \mathcal{P}_{h_l} = \mathcal{P}_{h_i} \\ 0.57 & \text{if } \text{ctf}(h_i) < 6.0 \ \& \ \mathcal{P}_{h_l} \neq \mathcal{P}_{h_i} \ \& \ \mathcal{S}_{h_l} = \mathcal{S}_{h_i} \\ 0.23 & \text{if } \text{ctf}(h_i) < 6.0 \ \& \ \mathcal{S}_{h_l} \neq \mathcal{S}_{h_i} \ \& \ \mathcal{G}_{h_l} = \mathcal{G}_{h_i} \\ 0.09 & \text{if } \text{ctf}(h_i) < 6.0 \ \& \ \mathcal{G}_{h_l} \neq \mathcal{G}_{h_i} \\ 0.0 & \text{otherwise} \end{cases}$$

Note that the design principle of these rewards is to reward more small microscope movement. As shown later in the experiments (Section 5.3), CryoRL is not sensitive to the changes of the rewards as long as the design described above is followed.

**Deep Q-learning.** We apply the deep Q-learning approach proposed in [22] to learn our policy for cryo-EM data collection. The goal of the agent is to select a sequence of actions (i.e. holes) based on a policy to maximize future rewards (i.e the total number of low-CTF holes). In Q-learning, this is achieved by maximizing the action-value function  $Q^*(s, a)$ , i.e. the maximum expected return achievable by any strategy (or policy)  $\pi$ , given an observation (or state)  $s$  and some action  $a$  to take. In other words,  $Q^*(s, a) = \max_{\pi} \mathbb{E}[R_t | s_t = s, a_t = a, \pi]$ , where  $R_t = \sum_{t'=t}^{\infty} \gamma^{t'-t} r_{t'}$  is the accumulated future rewards with a discount factor  $\gamma$ .  $Q^*$  can be found by solving the Bellman Equation [30] as follows,

$$Q^*(s, a) = \mathbb{E}_{s'}[r + \gamma \max_{a'} Q^*(s', a') | s, a] \quad (4)$$

In practice, the state-action space can be enormous, thus in [22], a deep neural network parameterized by  $\theta$  is applied to approximate the action-value function. The network is

referred to as Deep Q-Network (DQN) in the original paper. DQN can be trained by minimizing the following loss functions  $L(\theta)$ , which is often called Bellman error,

$$L(\theta) = \mathbb{E}_{s,a,r,s'}[(y - Q(s, a; \theta))^2] \quad (5)$$

where  $y = \mathbb{E}_{s'}[r(s, a) + \gamma \max_{a'} Q(s', a') | s, a]$  is the target for the current iteration. Experience replay is further adopted in [22] to store into memory the transition at each time-step, i.e  $(s_l, a_l, r_l, s_{l+1})$ , and then sample the stored samples for model update during training.

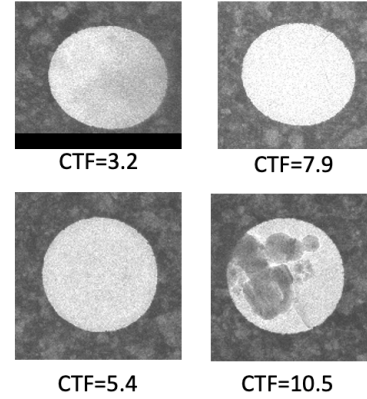


Figure 4. Examples of hole images and their CTF values. A hole with a CTF  $\leq 6$  is considered good in our paper.

**DQN with Action Elimination via Patch Ranking.** In a regular scenario such as playing Atari [22] where the action space is small and fixed, a network can be trained to predict all the actions at once. However, this is not suitable for our case as our action space is not fixed and can grow large depending on the training data size. To deal with this issue, we modify the Q-network to predict the Q-value for each hole (i.e., action) using one single output. The Q-value for all the actions can then be batch processed and the  $\epsilon$ -greedy scheme is applied for action selection. The DQN used in our work is a 3-layer fully connected network. The size of each layer is 128, 256 and 128, respectively.

The potentially enormously large action space in our problem makes policy exploration quite inefficient as most actions sampled from such a space are not useful. To avoid executing too many sub-optimal actions in learning, we propose an effective method to reduce the action space by restricting the valid actions to a small portion of holes predicted by the classifier as low-CTFs. We start by ranking all the grid-level images by their numbers of low-CTF predictions, from high to low, and then the patches in the same way. The high-ranked patches are likely to contain more valid holes and should be visited more frequently during the learning. The pre-specified duration (i.e.  $\tau$  in Eq. 1) as well as the switching costs  $t(h_l)$  defined in Section 4 allow us to obtain an upper limit  $N_{max}$  of good holes if *all the holes are assumed to be low CTF* and visited in the sorted order described above. We then select a minimum set of patches

Feature Type	Definition	Value
hole	is it low-CTF? is it visited?	{0,1}
patch/ square/grid	# of unvisited holes # of unvisited ICTFs # of visited holes # of visited ICTFs	0~150*
microscope movement	a new patch-level image? a new square-level image? a new grid-level image?	{0,1} {0,1} {0,1}

\*: the maximum number of holes allowed in a grid-level image in our setting

Table 1. Input features to DQN

$P$  with a total number of low-CTF predictions  $\geq \beta N_{max}$  ( $\beta > 0$ ), and all the holes in  $P$  define a reduced new space for Q-learning. Here,  $\beta$  is a user-defined parameter to control the size of the valid action set. Our approach is not sensitive to  $\beta$ , and any number between 1.0~2.5 works reasonably well. We thus empirically set  $\beta$  to 1.5 in test and a larger number 2.5 in training to encourage.

Unlike the Elimination Network (EAN) proposed in [41], our approach redefines the action space before Q-learning, so it can be applied to any policy learners without modification of them. We show later in the experiments that our approach results in a significant speedup of 2~3 times over the vanilla DQN and improves other policy learners such as A2C [7] and C51 [2] remarkably.

**Features to DQN.** The quality of a hole is directly determined by its CTF value. Similarly, the number of low-CTF holes (ICTFs) in a hole-level image indicates the quality (or value) of the image, and a good RL policy should always consider prioritizing high-quality patches first in planning. The same holds true for square-level and grid-level images. Based on this, we design hierarchical input features to the DQN according to the quality of images at different levels. We also consider the information of microscope movement as it tells whether the microscope is exploring a new region or staying at the same region. The details of these features can be found in Table 1. Finally, a sequence of these features for the last  $k - 1$  visited holes as well as the current one to be visited are concatenated together to form the input to DQN. In our experiments,  $k$  is empirically set to 4.

**Hole-level Classification.** We trained the hole-level classifier offline by cropping out the holes in our data using the location provided in the meta data. Fig. 4 illustrates a few examples of hole images. These images are actually visually ambiguous, confounding the task of building generalized hole classifiers, as shown in Section 5.2. Using an offline classifier enables fast learning of the Q function as only the Q-network is updated in training and its input features can be computed efficiently. However, it is possible to jointly learn the classifier and DQN to further improve performance. We leave this possibility for future work.

## 5. Experiments

### 5.1. Experimental Setup

**Dataset** To design and evaluate the performance of cryo-RL, we collected an "unbiased" cryo-EM dataset (Y1) to provide a systematic overview all squares, patches, holes, and micrographs within a defined region of a cryo-EM grid. Specifically, aldolase at a concentration of 1.6 mg/ml was dispensed on a support grid and prepared using a Vitrobot. Instead of picking the most promising squares and holes, we randomly selected 31 squares across the whole grid and imaged almost all the holes in these selected squares. This resulted in a dataset of 4017 micrographs from holes in these 31 squares. Overall, the data quality was poor, given that only 33.4% of the micrographs have a CTF below 6 Å. However, this makes the dataset very suitable for developing and testing algorithms for data collection algorithms, because 1) a perfect algorithm will aim to find the best data from mostly bad micrographs, and 2) the "unbiasedness" of this dataset ensures that when an algorithm selects a hole, the corresponding micrograph, and its metric can be provided as feedback.

In addition, we collected another different dataset (Y2) of 3969 micrographs with a different sample and grid type. We split both datasets into training and validation sets by a ratio of 2:1. In the experiments below, we evaluate our approach mainly based on Y1 while using Y2 to test the transferability of CryoRL.

**Training and Evaluation.** We used the Tianshou reinforcement learning framework [36] to learn cryoRL. Each model was trained with 20 epochs, using the Adam optimizer and an initial learning rate of 0.01. We set the duration in our system to 240 minutes for training, and evaluate the system at 120, 240, 360 and 480 minutes, respectively.

### 5.2. Main Results

**Comparison with Baselines.** We first developed a greedy-based method purely based on the hole classification results. This method performs a primary sorting on the grid-level images by their quality (i.e., the total number of low CTF holes), followed by a secondary sorting on the patches within a grid by the quality of patches. The sorted patches are then scanned in order, with only the holes classified as low CTFs visited. While simple, this greedy approach serves as a strong baseline when the hole-level classifier is strong.

We also compare our approach with two other widely used optimization techniques in practice: Genetic Algorithms (GA) [35] and Simulated Annealing (SA) [12]. In these two solvers, solutions are sampled at the patch level rather than at the hole level for efficiency, and the fitness of the solutions are assessed according to the objective function proposed in this paper, i.e Eq. 1. Since GA and SA

are largely based on heuristic, the best solutions determined by them are scanned in a similar way to the greedy-based approach described above during the evaluation.

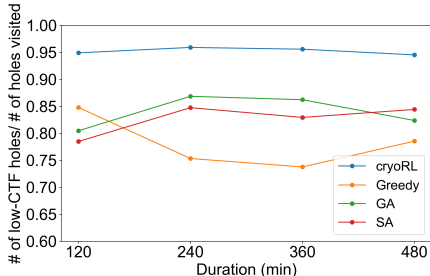


Figure 5. Percentage of ICTF images visited during data collection.

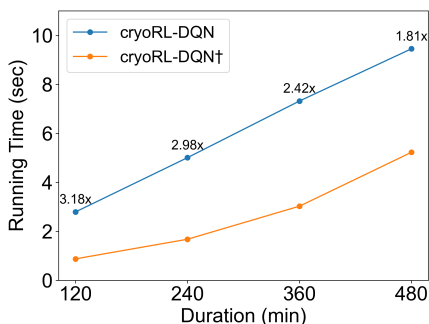


Figure 6. Runtime comparison of CryoRL-DQN and its fast version with action elimination (CryoRL-DQN<sup>†</sup>).

Methods	$\tau=120$	$\tau=240$	$\tau=360$	$\tau=480$
Random	2.6 $\pm$ 1.4	5.1 $\pm$ 1.6	7.3 $\pm$ 2.3	9.8 $\pm$ 2.2
Greedy	41.8 $\pm$ 2.5	69.3 $\pm$ 3.2	104.9 $\pm$ 4.9	147.9 $\pm$ 5.1
Genetic Alg. (GA) [35]	28.3 $\pm$ 6.5	72.3 $\pm$ 6.8	115.7 $\pm$ 7.8	150.4 $\pm$ 6.8
Simulated Annealing (SA) [12]	39.4 $\pm$ 6.5	73.3 $\pm$ 7.0	104.7 $\pm$ 8.9	147.9 $\pm$ 9.6
offline path planing [30]	44.3 $\pm$ 0.9	84.6 $\pm$ 6.1	121.4 $\pm$ 6.7	166.6 $\pm$ 4.9
CryoRL-DQN (ours)	41.7 $\pm$ 3.1	86.6 $\pm$ 3.0	132.0 $\pm$ 2.3	171.4 $\pm$ 2.0
CryoRL-DQN <sup>†</sup> (ours)	47.4 $\pm$ 0.5	89.0 $\pm$ 3.1	131.8 $\pm$ 1.8	172.6 $\pm$ 2.0
human	31.9 $\pm$ 10.6	77.4 $\pm$ 6.2	-	-

Table 2. Comparison of CryoRL with baseline approaches on Y1

Table 2 reports the total number of low-CTF holes (#ICTF) found by each approach. For fair comparison, all the results are averaged over 50 trials starting from random picked holes. Here, ResNet50 is used as the offline classifier, which achieves an accuracy around 83% in low-CTF hole classification (see Table 4). The results based on ResNet18 can be found in the supplemental materials. As shown in the table, our approach (cryoRL-DQN) is clearly superior to all the baseline methods, producing quite promising results. With action elimination, the fast version of CryoRL (cryoRL-DQN<sup>†</sup>) improve the performance further. Note that while offline path planning yields comparable performance to our method, it is prohibitively costly in computation.

To further illustrate the advantage of our approach, we plot for each approach the percentage of low-CTF holes over the total number of holes visited by time in Fig. 5.

Our approach demonstrates high efficacy in data collection, finding  $\sim 95\%$  of the holes in good quality. As a comparison, the percentage of low-CTF holes in Y1 is 33.4% and the classification accuracy of low CTFs is only 83.9%.

We also experimented with several other RL variants including dueling DQN [33], DQN with prioritized replay [26], A2C [7] and C51 [2]. As seen from Table 3, the DQN family overall perform better than A2C and C51. Interestingly, A2C and C51 benefit substantially from action elimination and gain significant performance boosts, suggesting that restricting the actions to smaller valid sets helps these methods learn policies more effectively. CryoRL with action elimination also achieves considerable speedups in runtime by 2 $\sim$ 3 times, as shown in Fig. 6. Since the performance differences between DQN models are minor, we focus on the vanilla DQN in the analysis below.

**Comparison with Human Performance.** We developed a simulation tool to benchmark human performance against the performance of cryoRL. Fifteen students from two different cryo-EM labs with various expertise levels were recruited in this human study. The users did not have any prior knowledge of this specific dataset before participating in this study. Patch images containing holes in the same dataset were shown to the user. The user had either 50 or 100 chances to select the holes to take micrographs from, corresponding to the experiment’s test duration of 120 or 240 minutes. After each selection, the CTF value for the selected hole was provided to the user. The goal of the users is to select as many “good” holes as possible in 50 or 100 chances. Note that we did not penalize the users for switching to a different patch or square as we did in cryoRL. This encouraged the users to explore different patches initially and, theoretically, resulting in better performance than penalties applied. Nevertheless, we found that cryoRL outperforms the human performance in both time durations (Table 2).

**Transferrability.** We further evaluate the *transferability* of our proposed approach based on a new dataset **M**, which consists of users’ daily use of microscope in real-life scenarios from 2019 to 2021. Different from Y1 and Y2 data where almost each hole in the patch images was imaged, **M** were only sparsely inspected, with a small portion of holes visited by the users. In other words, there are a lot of holes in the patch images without a CTF ground truth available. As a result, the limited coverage in **M** data is not sufficient for learning effective RL policies for planning microscope movement. Nevertheless, **M** data were collected under different realistic settings where various grid types and microscopes were used. It is much more diverse and substantially larger than Y1 data (over 100,000 holes with CTF ground truth in **M** vs. 4,000 in **Y**), making it suitable for building a foundation model for hole classification.

We split **M** data into training and validation sets at a ratio

Methods	$\tau=120$	$\tau=240$	$\tau=360$	$\tau=480$
CryoRL-A2C	35.5±7.8	74.0±9.0	111.3±8.8	147.0±8.8
CryoRL-CS1	39.4±4.2	76.3±3.1	109.6±2.0	141.0±2.7
CryoRL-DQN	41.7±3.1	86.6±3.0	132.0±2.3	171.4±2.0
CryoRL-DQN (dueling)	44.6±3.3	89.3±4.4	126.3±4.2	157.4±4.4
CryoRL-DQN (prioritized)	42.5±4.3	86.4±3.9	128.7±5.1	172.0±3.5
CryoRL-A2C <sup>†</sup>	47.0±1.3(+32.3%)	90.8±4.0(+22.7%)	128.2±2.4(+15.1%)	163.9±4.7(+11.5%)
CryoRL-CS1 <sup>†</sup>	47.4±0.9(+20.3%)	82.2±2.3(+7.7%)	116.9±1.0(+6.7%)	144.0±1.9(+2.1%)
CryoRL-DQN <sup>†</sup>	<b>47.4±0.5(+13.7%)</b>	89.0±3.1(+2.8%)	131.8±1.8(+0.0%)	172.6±2.0(+1.0%)
CryoRL-DQN <sup>†</sup> (dueling)	47.3±1.0(+6.1%)	89.4±2.9(+0.0%)	128.6±2.0(+1.8%)	165.4±2.5(+5.1%)
CryoRL-DQN <sup>†</sup> (prioritized)	47.2±1.7(+11.1%)	<b>90.6±2.8(+4.9%)</b>	<b>132.9±3.0(+3.3%)</b>	<b>174.0±3.1(+1.2%)</b>

Table 3. Perf. comparison of different CryoRL variants on Y1 (<sup>†</sup> indicates action elimination (Section 4.3)). The performance gains from action elimination are highlighted by numbers in parentheses.

Test	Training		Top1 Acc.			#ICTFs found			
	classifier	CryoRL	ICTF	hCTF	all	$\tau=120$	$\tau=240$	$\tau=360$	$\tau=480$
Y1	Y1	Y1	83.9	91.2	88.5	47.4±0.5	89.0±3.1	131.8±1.8	172.6±2.0
Y1	M	Y1	66.6	85.1	73.6	44.7±2.2	70.0±4.7	104.0±3.6	138.9±2.6
Y2	M	Y2	69.5	77.4	73.5	31.0±6.5	56.3±7.7	87.1±8.0	125.5±8.3
Y2	M	Y1	69.5	77.4	73.5	20.9±4.3	55.8±3.7	83.1±3.5	91.8±3.3

Table 4. Generalization ability of the offline classifier and CryoRL.

of 4:1 and trained a hole classifier based on Resnet50. We then applied the classifier to both test sets in Y1 and Y2, and the results are listed in Table 4. As seen from the table, the classifier achieves moderate performance on Y1 and Y2, with an accuracy of around 70% in low-CTF classification, suggesting that hole classification is still a challenging problem that needs further improvement.

We further trained RL models on Y1 and Y2 using the classification results based on the M model mentioned above. As shown in Table 4, a modest classifier (M) results in a performance drop in CryoRL (4<sup>th</sup> row) as expected, but the results are still reasonably good. Additionally, we extend to test the transferability of the RL models. Specifically, we applied the RL model based on Y1 to Y2 dataset and compared the results (6<sup>th</sup> row) to those from the RL model trained on Y2 itself (5<sup>th</sup> row). Even though Y1 and Y2 datasets were collected with different samples and grid types, the results between these two models are still comparable, showing the good transferability of CryoRL.

### 5.3. Ablation Study

In this section, we conduct experiments to characterize our proposed approach. We investigate how hole time duration and rewarding affects the performance of cryoRL (i.e. the total number a low-CTF holes in a given amount of time).

**Effects of Time Duration.** In principle, the time duration  $\tau$  used in training cryoRL controls the degree of interaction of the RL agent with the data. A small  $\tau$  limits cryoRL to a few high-quality patches only, which might result in a more conservative policy that underfits. We confirm this potential issue, showing inferior performance when a short duration of 120 minutes is used for training.

**Effects of Rewarding Strategies.** In our approach, the rewards used in policy learning are empirically determined. To check the potential impact of different rewards on the performance of cryoRL, we trained more Q networks by doubling the reward for a) square switching; b) grid switching; and c) both. These changes are intended to encourage more active exploration of the data. The different rewarding schemes perform comparably, and increasing the reward for square switching leads to slightly better performance than the default setting. This suggests that CryoRL is not sensitive to rewards setup as long as the rewards value small microscope movement more. Nevertheless, how to optimize rewards for better performance of cryoRL is an area of improvement in future work.

## 6. Conclusion

To summarize, by combining supervised classification and deep RL, cryoRL provides a new framework for cryo-EM data collection. It can not only return the quality predictions for lower magnified hole level images but can also plan the trajectory for data acquisition. We have shown that cryoRL combined with an offline hole classifier achieves better performance than average human users. Nevertheless, cryoRL needs squares to be pre-selected and all their corresponding patch-level images to be pre-captured. Future work will be needed to further optimize the RL system to consider more of this hierarchical process of cryo-EM data collection. The specific hyper-parameters, especially the penalties in the reward function, can also be improved for a more practical application.



## References

- [1] Philip R Baldwin, Yong Zi Tan, Edward T Eng, William J Rice, Alex J Noble, Carl J Negro, Michael A Cianfrocco, Clinton S Potter, and Bridget Carragher. Big data in cryoem: automated collection, processing and accessibility of em data. *Current opinion in microbiology*, 43:1–8, 2018.
- [2] Marc G Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. In *International Conference on Machine Learning*, pages 449–458. PMLR, 2017.
- [3] Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Debiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680*, 2019.
- [4] Anders Brahma. *Comprehensive biomedical physics*. Newnes, 2014.
- [5] Jack PK Bravo, Tyler L Dangerfield, David W Taylor, and Kenneth A Johnson. Remdesivir is a delayed translocation inhibitor of sars-cov-2 replication. *Molecular cell*, 81(7):1548–1552, 2021.
- [6] Anchi Cheng, Edward T Eng, Lambertus Alink, William J Rice, Kelsey D Jordan, Laura Y Kim, Clinton S Potter, and Bridget Carragher. High resolution single particle cryo-electron microscopy using beam-image shift. *Journal of structural biology*, 204(2):270–275, 2018.
- [7] P. Dhariwal, C. Hesse, O. Klimov, A. Nichol, M. Plappert, A. Radford, J. Schulman, S. Sidor, Y. Wu, and P. Zhokhov. Openai baselines. 2017.
- [8] Ahmed Fawzy Gad. Pygad: An intuitive genetic algorithm python library, 2021.
- [9] Eric Hand. ‘we need a people’s cryo-em.’ scientists hope to bring revolutionary microscope to the masses. *Science*, 2020.
- [10] Anssi Kanervisto, Christian Scheller, and Ville Hautamäki. Action space shaping in deep reinforcement learning. In *2020 IEEE Conference on Games (CoG)*, pages 479–486. IEEE, 2020.
- [11] Zunlong Ke, Joaquin Oton, Kun Qu, Mirko Cortese, Vojtech Zila, Lesley McKeane, Takanori Nakane, Jasenko Zivanov, Christopher J Neufeldt, Berati Cerikan, et al. Structures and distributions of sars-cov-2 spike proteins on intact virions. *Nature*, 588(7838):498–502, 2020.
- [12] Scott Kirkpatrick. Optimization by simulated annealing: Quantitative studies. *Journal of statistical physics*, 34(5):975–986, 1984.
- [13] Goran Kokic, Hauke S Hillen, Dimitry Teginov, Christian Dienemann, Florian Seitz, Jana Schmitzova, Lucas Farnung, Aaron Siewert, Claudia Höbartner, and Patrick Cramer. Mechanism of sars-cov-2 polymerase stalling by remdesivir. *Nature communications*, 12(1):1–7, 2021.
- [14] Thomas Kollar and Nicholas Roy. Trajectory optimization using reinforcement learning for map exploration. *The International Journal of Robotics Research*, 27(2):175–196, 2008.
- [15] Alexander Krull, Peter Hirsch, Carsten Rother, Augustin Schiffrin, and C Krull. Artificial-intelligence-driven scanning probe microscopy. *Communications Physics*, 3(1):1–8, 2020.
- [16] Gabriel C Lander, Scott M Stagg, Neil R Voss, Anchi Cheng, Denis Fellmann, James Pulokas, Craig Yoshioka, Christopher Irving, Anke Mulder, Pick-Wei Lau, et al. Appion: an integrated, database-driven pipeline to facilitate em image processing. *Journal of structural biology*, 166(1):95–102, 2009.
- [17] Alessandro Lazaric, Marcello Restelli, and Andrea Bonarini. Reinforcement learning in continuous action spaces through sequential monte carlo methods. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2008.
- [18] Ngan Le, Vidhiwar Singh Rathour, Kashu Yamazaki, Khoa Luu, and Marios Savvides. Deep reinforcement learning in computer vision: A comprehensive survey. *CoRR*, abs/2108.11510, 2021.
- [19] Kywoon Lee, Sol-A Kim, Jaesik Choi, and Seong-Whan Lee. Deep reinforcement learning in continuous action spaces: a case study in the game of simulated curling. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2937–2946. PMLR, 10–15 Jul 2018.
- [20] Florian A Lempp, Leah B Soriaga, Martin Montiel-Ruiz, Fabio Benigni, Julia Noack, Young-Jun Park, Siro Bianchi, Alexandra C Walls, John E Bowen, Jiayi Zhou, et al. Lectins enhance sars-cov-2 infection and influence neutralizing antibodies. *Nature*, 598(7880):342–347, 2021.
- [21] Yilai Li, Jennifer N Cash, John JG Tesmer, and Michael A Cianfrocco. High-throughput cryo-em enabled by user-free preprocessing routines. *Structure*, 28(7):858–869, 2020.
- [22] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning, 2013.

- [23] Mohammadreza Nazari, Afshin Oroojlooy, Lawrence Snyder, and Martin Takác. Reinforcement learning for solving the vehicle routing problem. *Advances in neural information processing systems*, 31, 2018.
- [24] Matthew Perry. Simanneal: <https://github.com/perrygeo/simanneal>, 2020.
- [25] Alexis Rohou and Nikolaus Grigorieff. Ctfnd4: Fast and accurate defocus estimation from electron micrographs. *Journal of structural biology*, 192(2):216–221, 2015.
- [26] Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, 2015.
- [27] Johannes F Scheid, Christopher O Barnes, Basak Eraslan, Andrew Hudak, Jennifer R Keefe, Lisa A Cosimi, Eric M Brown, Frauke Muecksch, Yiska Weisblum, Shuting Zhang, et al. B cell genomics behind cross-neutralization of sars-cov-2 variants and sars-cov. *Cell*, 2021.
- [28] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneshelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, jan 2016.
- [29] Markus Stabrin, Fabian Schoenfeld, Thorsten Wagner, Sabrina Pospich, Christos Gatsogiannis, and Stefan Raunser. Transpire: automated and feedback-optimized on-the-fly processing for cryo-em. *Nature communications*, 11(1):1–14, 2020.
- [30] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.
- [31] Dmitry Tegunov and Patrick Cramer. Real-time cryo-electron microscopy data preprocessing with warp. *Nature methods*, 16(11):1146–1152, 2019.
- [32] Peng Tong, Juan Liu, Xijun Wang, Bo Bai, and Huaiyu Dai. Deep reinforcement learning for efficient data collection in uav-aided internet of things. In *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, pages 1–6. IEEE, 2020.
- [33] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado Hasselt, Marc Lanctot, and Nando Freitas. Dueling network architectures for deep reinforcement learning. In *International conference on machine learning*, pages 1995–2003. PMLR, 2016.
- [34] Felix Weis and Wim JH Hagen. Combining high throughput and high quality for cryo-electron microscopy data collection. *Acta Crystallographica Section D: Structural Biology*, 76(8):724–728, 2020.
- [35] Thomas Weise. Global optimization algorithms-theory and application. *Self-Published Thomas Weise*, 361, 2009.
- [36] Jiayi Weng, Huayu Chen, Dong Yan, Kaichao You, Alexis Duburcq, Minghao Zhang, Hang Su, and Jun Zhu. Tianshou: A highly modularized deep reinforcement learning library. *arXiv preprint arXiv:2107.14171*, 2021.
- [37] Daniel Wrapp, Nianshuang Wang, Kizzmekia S Corbett, Jory A Goldsmith, Ching-Lin Hsieh, Olubukola Abiona, Barney S Graham, and Jason S McLellan. Cryo-em structure of the 2019-ncov spike in the pre-fusion conformation. *Science*, 367(6483):1260–1263, 2020.
- [38] Hangping Yao, Yutong Song, Yong Chen, Nanping Wu, Jialu Xu, Chujie Sun, Jiaying Zhang, Tianhao Weng, Zheyuan Zhang, Zhigang Wu, et al. Molecular architecture of the sars-cov-2 virus. *Cell*, 183(3):730–738, 2020.
- [39] Deheng Ye, Zhao Liu, Mingfei Sun, Bei Shi, Peilin Zhao, Hao Wu, Hongsheng Yu, Shaojie Yang, Xipeng Wu, Qingwei Guo, et al. Mastering complex control in moba games with deep reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6672–6679, 2020.
- [40] Wanchao Yin, Chunyou Mao, Xiaodong Luan, Dandan Shen, Qingya Shen, Haixia Su, Xiaoxi Wang, Fulai Zhou, Wenfeng Zhao, Minqi Gao, et al. Structural basis for inhibition of the rna-dependent rna polymerase from sars-cov-2 by remdesivir. *Science*, 368(6498):1499–1504, 2020.
- [41] Tom Zahavy, Matan Haroush, Nadav Merlis, Daniel J Mankowitz, and Shie Mannor. Learn what not to learn: Action elimination with deep reinforcement learning. *Advances in Neural Information Processing Systems*, 31, 2018.
- [42] Yu Zhang, Zhiyu Mou, Feifei Gao, Ling Xing, Jing Jiang, and Zhu Han. Hierarchical deep reinforcement learning for backscattering data collection with multiple uavs. *IEEE Internet of Things Journal*, 8(5):3786–3800, 2020.
- [43] Xiangyu Zhao, Liang Zhang, Zhuoye Ding, Long Xia, Jiliang Tang, and Dawei Yin. Recommendations with Negative Feedback via Pairwise Deep Reinforcement Learning. In *Proceedings of the 24th International Conference on Knowledge Discovery and Data Mining*, pages 1040–1048. ACM, 2018.

- [44] Ang Zhu, Varun Gupta, Satyajeet Singh Ahuja, Yuan-dong Tian, Ying Zhang, and Xin Jin. Network Planning with Deep Reinforcement Learning. In *SIGCOMM*. ACM, 2021.