2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Tactile Active Inference Reinforcement Learning for Efficient Robotic Manipulation Skill Acquisition

Zihao Liu†, Xing Liu†, Yizhai Zhang, Zhengxiong Liu and Panfeng Huang*

Abstract— **Robotic manipulation holds the potential to replace humans in the execution of tedious or dangerous tasks. However, control-based approaches are not suitable due to the difficulty of formally describing open-world manipulation in reality, and the inefficiency of existing learning methods. Therefore, applying manipulation in a wide range of scenarios presents significant challenges. In this study, we propose a novel framework for skill learning in robotic manipulation called Tactile Active Inference Reinforcement Learning (Tactile-AIRL), aimed at achieving efficient learning. To enhance the performance of reinforcement learning (RL), we introduce active inference, which integrates model-based techniques and intrinsic curiosity into the RL process. This integration improves the algorithm's training efficiency and adaptability to sparse rewards. Additionally, we have designed universal tactile static and dynamic features based on vision-based tactile sensors, making our framework scalable to many manipulation tasks learning involving tactile feedback. Simulation results demonstrate that our method achieves significantly high training efficiency in objects pushing tasks. It enables agents to excel in both dense and sparse reward tasks with just few interaction episodes, surpassing the SAC baseline. Furthermore, we conduct physical experiments on a gripper screwing task using our method, which showcases the algorithm's rapid learning capability and its potential for practical applications.**

I. INTRODUCTION

Currently, robotic manipulation mostly relies on programmed instructions for control. Developers use their knowledge to design procedures and form task plans, and robots perform skilled behaviors based on position and force information [1]. However, this approach faces two main challenges: unstructured real-world environments are difficult to describe with standardized shapes, and human knowledge can't predict all situations robots may encounter, limiting the application of robotic manipulation to well-defined environments. To address these challenges, tactile perception and reinforcement learning (RL) are used to improve scene understanding and reduce policy blind spots respectively. Tactile information provides a more detailed scene description than pose information alone, and RL adapts to various state configurations through exploration. However, traditional tactile sensing is sparse [2], and RL is data-inefficient.

This work was supported in part by the National Key R&D Program of China under Grant 2022ZD0117900, and the National Natural Science Foundation of China under Grant 62103334, 92370123, 62273280, 62103337.

Zihao Liu, Xing Liu, Yizhai Zhang, Zhengxiong Liu, and Panfeng Huang(Corresponding author) are with the Research Center for Intelligent Robotics, School of Astronautics, Northwestern Polytechnical University, and National Key Laboratory of Aerospace Flight Dynamics, Northwestern Polytechnical University, Xi'an, China, 710072 e-mail: pfhuang@nwpu.edu.cn, xingliu@nwpu.edu.cn.

Unlike commonly used force / torque sensors or strain-based tactile sensors, vision-based tactile sensors such as GelSight [3] offer a new approach to obtain tactile information by deforming a gel that contacts an object. In this work, we explore the integration of vision-based tactile sensors in robot manipulation skill learning to enhance learning efficiency. Specifically, suitable handcrafted image features will be designed and integrated into the state space, enhancing performance and reducing computational demands.

On the other hand, to address the data efficiency issue of vanilla RL, model-based RL [4] has been widely employed. Typically, model-based methods learn a state transition model of the agent's environment and use the model for planning and action execution, improving the utilization efficiency of sampled data. However, accumulated errors due to model can degrade performance in sparse reward contexts. Thus, combining exploration with exploitation is essential, enhancing adaptability to sparse rewards and simplifying reward function design. And, integrating tactile sensors with this RL approach improves the feasibility of learning robotic manipulation in real-world scenarios.

To summarize our contributions in this work:

- We design a method to integrate vision-based tactile information into reinforcement learning in the context of robotic manipulation, thereby enhancing the perception of manipulation tasks.
- We introduce Active Inference Reinforcement Learning to acquire robotic manipulation skills, aiming to enhance the data efficiency of learning manipulation skills.
- We propose the Tactile-AIRL framework and validate its effectiveness through simulation and real-world experiments. In simulations, the manipulator learns to push objects up a slope using tactile sensors. In real-world experiments, we employ a gripper to learn the twisting of nuts to minimize slippage. The results demonstrate that our algorithm strikes a balance between exploration and exploitation, achieving data efficiency that exceeds that of other algorithms.

II. RELATED WORK

A. Vision-based Tactile Sensor

Tactile sensors are used to digitize contact signals in the physical world. In recent years, vision-based tactile sensors such as the gelsight digit tactile sensor [3] have emerged. These sensors visualize the sensor deformation of the contact surface by using a camera to convert touch into vision, such as Figure 3 illustrates the tactile image during manipulation.

Due to their high resolution and rich information, this type of sensor has unique advantages and has been successfully applied to various robot operation tasks, such as cutting [5], dish loading [6], and robotic manipulation [7].

For vision-based tactile sensor, poisson reconstruction [8] can be used to recover precise depth images of its contact surface, such as for evaluation of the pose and force of grasping cables [7]. In addition, using neural networks to process the images from tactile sensors has become a popular approach. Feature extraction using neural networks can handle more complex contact [9] and perform surface reconstruction of objects [10].

B. Reinforcement Learning and Active Inference

Many works integrate reinforcement learning (RL) into robotics to develop intelligent robots. However, RL is data inefficient, requiring sim2real techniques or improved RL performance to operate effectively with limited real-world data. Regarding sim2real, quadruped robot could train extensively on motion behavior in simulation and then transfer to the real world [11]. Meanwhile, half sim2real [12] reduces the amount of training required on the real world by treating simulation as a pre-training parameter process. In terms of improving the performance of RL itself, VPG [13] achieves unstructured object placement through the selection of several predefined skills executed by the robotic arm on image pixels, where the predefined skills reduce the algorithm's exploration space, and accelerate training speed. RHER [14] improves data utilization through hierarchical thinking and HER methods, which can be used for sparse rewards from judging the success or failure of the task.

Active inference [15] arises from the free energy principle[16]. Originally used to describe behavioral motivation in biological organisms, the free energy principle states that organisms tend to spontaneously decrease their free energy while operating. To simulate this process of decreasing free energy, active inference is used for variational optimization. And active inference can be naturally extended to robots that have actuators. For example, active inference adaptive control [17] does not require knowledge of the robot's geometric structure and can achieve smooth multi-joint control during the dynamic reduction of free energy. An approach related to our work is the Active Inference Reinforcement Learning (AIRL) framework [18], which interprets active inference in the context of RL, where free energy includes both modeling accuracy and reward.

III. METHOD

Tactile-AIRL is a framework that improves learning efficiency in robotic manipulation by integrating tactile information and applying active inference principles. The workflow of Tactile-AIRL is shown in Figure 1.

A. Active Inference Reinforcement Learning

The active inference of the agent aims to minimize its own free energy during action, leading to a more precise understanding of the agent's environment. To accomplish this, we employ the decision-making scheme known as Free Energy of the Expected Future (FEEF) [18]. By considering the minimization of current and future free energy, this approach transforms the task into a planning problem.

Let the variable $x_{t:T}$ denote a sequence of variables over time, $x_{t:T} = x_t, \cdots, x_T$. Here, θ represents the parameters of the neural networks, and π represents the policy. Next, we define the concatenation of o_{p_i} and o_{t_i} as o_i, and consider the reward as a partial observation. Furthermore, $q(r_{t:T}, o_{t:T}, \theta, \pi)$ represents the beliefs of a robot regarding future variables, while $p^{\Phi}(r_{t:T}, o_{t:T}, \theta)$ represents a biased generative model for the robot. The FEEF $\widetilde{\mathcal{F}}$ that needs to be minimized is defined as follows:

$$\widetilde{\mathcal{F}} = D_{KL}\Big(q(r_{0:T}, o_{0:T}, \theta, \pi)||p^{\Phi}(r_{0:T}, o_{0:T}, \theta)\Big) \tag{1}$$

Noting that the FEEF contains the policy π, we will minimize the FEEF by adjusting the policy $q(\pi)$. After some derivation, we can get:

$$\widetilde{\mathcal{F}} = 0 \Rightarrow D_{KL}\Big(q(\pi)||e^{-\widetilde{\mathcal{F}}_{\pi}}\Big) = 0 \tag{2}$$

where

$$\widetilde{\mathcal{F}}_{\pi} = D_{KL}\Big(q(r_{0:T}, o_{0:T}, \theta|\pi)||p^{\Phi}(r_{0:T}, o_{0:T}, \theta)\Big) \tag{3}$$

Therefore, we can fit the distribution $q(\pi)$ to $e^{-\widetilde{\mathcal{F}}_{\pi}}$ to obtain a policy that minimizes free energy. Consequently, the problem at hand exhibits similarities with model-based reinforcement learning. By utilizing a planning algorithm to minimize the future $\widetilde{\mathcal{F}}_{\pi}$, we can derive a near-optimal policy. And minimizing $\widetilde{\mathcal{F}}_{\pi}$ holds practical significance. To simplify the notation, let's temporarily abbreviate the sequences $r_{0:T}$ and $o_{0:T}$ as r and s, respectively. We can decompose $\widetilde{\mathcal{F}}_{\pi}$ into two terms: the expected information gain term and the extrinsic term, as follows:

$$\begin{aligned}
-\widetilde{\mathcal{F}}_{\pi} \approx & \mathbb{E}_{q(o,\theta|r,\pi)q(r|\pi)}\Big[\ln p(o,\theta|r,\pi) - \ln q(o,\theta|\pi)\Big] \\
& - \mathbb{E}_{q(r|o,\theta,\pi)q(o,\theta|\pi)}\Big[\ln q(r|o,\theta,\pi) - \ln p^{\Phi}(r)\Big] \\
= & \underbrace{\mathbb{E}_{q(r|\pi)}\Big[D_{KL}\Big(q(o,\theta|r,\pi)||q(o,\theta|\pi)\Big)\Big]}_{\text{expected information gain term, c}} \\
& - \underbrace{\mathbb{E}_{q(o,\theta|\pi)}\Big[D_{KL}\Big(q(r|o,\theta,\pi)||p^{\Phi}(r)\Big)\Big]}_{\text{extrinsic term, r}}
\end{aligned} \tag{4}$$

Thus, the minimization of $\widetilde{\mathcal{F}}_{\pi}$ is equivalent to the maximization of expected information gain, indicating a preference for observations that provide new information. This intrinsic curiosity resembles mutual information [19]. Simultaneously, minimizing $\widetilde{\mathcal{F}}_{\pi}$ also reduces the extrinsic term, with the aim of bringing future observations closer to the prior setting. In the context of AIRL, this preference for priors is achieved through the expected value of the reward function. Consequently, the process of minimizing $\widetilde{\mathcal{F}}_{\pi}$ serves a dual role in both exploration and exploitation.

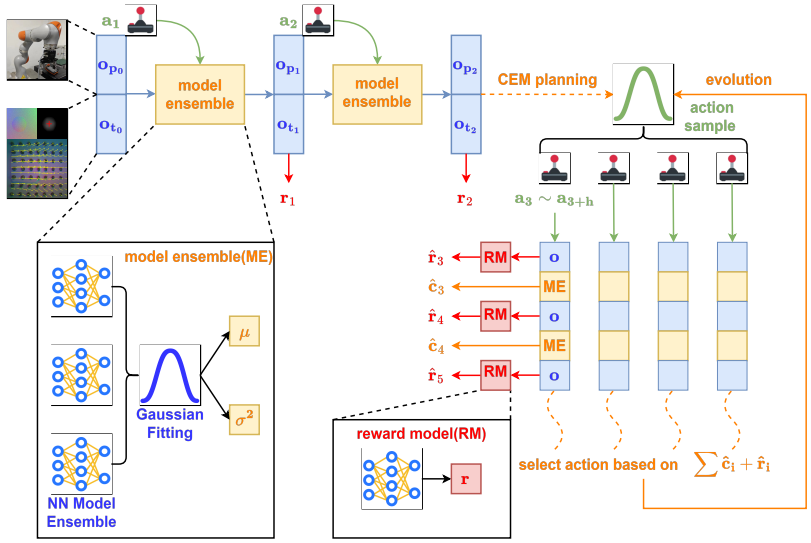

Fig. 1. The framework of Tactile Active Inference Reinforcement Learning. Here, o_{p_i} represents the state vector of the manipulator and objects, o_{t_i} represents tactile features, o is the abbreviation for the combination of o_{t_i} and o_{p_i}, a_i denotes the action, h denotes the planning horizon, and r_i denotes the reward generated by the environment. The **model ensemble** and the **reward model** refer to the neural network models to be learned. And $\hat{r}_i$ is the predict reward generated by reward model, $\hat{c}_i$ denotes the state information gain and is also a curiosity item, which is estimated by the model ensemble. The CEM algorithm here uses the model ensemble and the reward model to perform a lot of guessing and selects the best action sequence after evolutions.

More precisely, the execution of AIRL can be divided into three steps: evaluating the future states, evaluating $\widetilde{\mathcal{F}}_\pi$, and learning the policy.

1) Evaluating the Future States: The reason for evaluating future states is that calculating $\widetilde{\mathcal{F}}_\pi$ requires a distribution of future states over a period of time, which can be predicted from the current time using the trained state transition ensemble model:

$$\begin{aligned} q\left(o_{t:T}, r_{t:T}, \theta \mid \pi\right) &= p(\theta) \hookleftarrow \\ &\hookrightarrow \prod_{\tau=t}^{T} q\left(r_\tau \mid o_\tau, \theta, \pi\right) q\left(o_\tau \mid o_{\tau-1}, \theta, \pi\right) \\ q\left(r_\tau \mid o_\tau, \theta, \pi\right) &= \mathbb{E}_{q(o_\tau \mid \theta, \pi)}\left[p\left(r_\tau \mid o_\tau\right)\right] \\ q\left(o_\tau \mid o_{\tau-1}, \theta, \pi\right) &= \mathbb{E}_{q(o_{\tau-1} \mid \theta, \pi)}\left[p\left(o_\tau \mid o_{\tau-1}, \theta, \pi\right)\right] \end{aligned} \tag{5}$$

2) Evaluating $\widetilde{\mathcal{F}}_\pi$: Estimating $\widetilde{\mathcal{F}}_\pi$ involves evaluating the state trajectory generated by a specific action sequence during action sampling. Subsequently, the trajectory with a lower free energy will be selected. After some derivation, $\widetilde{\mathcal{F}}_\pi$ at each time step is calculated as follows. The summation of these values in future steps can be regarded as the value of future actions.

$$\begin{aligned} -\widetilde{\mathcal{F}}_{\pi_\tau} \approx & -\mathbb{E}_{q(o_\tau, \theta \mid \pi)}\Big[D_{KL}\Big(q(r_\tau \mid o_\tau, \theta, \pi) \| p^{\Phi}(r_\tau)\Big)\Big] \\ & + \underbrace{\mathbf{H}\Big[\mathbb{E}_{q(\theta)}[q(o_\tau \mid o_{\tau-1}, \theta, \pi)]\Big] - \mathbb{E}_{q(\theta)}\Big[\mathbf{H}[q(o_\tau \mid o_{\tau-1}, \pi, \theta)]\Big]}_{\text{state information gain, c}} \end{aligned} \tag{6}$$

3) Learning Policy: Because we can only obtain a score (free energy) corresponding to a certain sampled action sequence, optimization-based planning methods are not appropriate. In this work, we choose the sampling-based CEM planning algorithm. The iterative target value is $-\widetilde{\mathcal{F}}_\pi$, which is represented numerically as $e^{-\widetilde{\mathcal{F}}_\pi}$. At each action execution, a Gaussian distribution of action sequences is initialized as the initial value for planning actions. Then, the Gaussian distribution parameters are updated by sampling actions and evaluating $\widetilde{\mathcal{F}}_\pi$. Once the parameters of the Gaussian distribution stabilize, the first action in planning is taken as the executed action.

B. Tactile Information

Vision-based tactile sensors enhance manipulation tasks by integrating tactile information into the state space of reinforcement learning, leading to superior performance compared to using manipulator pose alone. Two methods exist for extracting tactile information from these sensors: hand-crafted features and deep neural networks. Our experiments show that using reconstruction loss for tactile images in multi-step predictions impairs reinforcement learning, likely due to the image format's limited information. Therefore, in this work, we opt for hand-crafted image features as o_t and concatenate them into the state space of the robot. Besides, to fully capture tactile sensations, we classify tactile features into static (contact location, intensity, shape) and dynamic (object motion trends like sliding).

Considering a fixed contact shape, the contact location and intensity can describe the static features of an object when contact. And, They can be estimated based on the centroid position and pixel summation of the contact surface depth image, which can be obtained by applying the Poisson integral to vertical surface gradients of RGB images [20]. For an object with a regular-shaped surface, tactile depth images usually consist of single connected areas. Therefore, the features calculation can be performed on the complete depth

images. In this study, we focus on this simple case. Suppose the depth image's (i,j)-th raw moments of the area to be calculated are denoted as m_{ij}. The centroid and summation can be obtained as follows:

$$\mu = (m_{10}/m_{00}, m_{01}/m_{00}) \tag{7}$$
$$\Sigma = m_{00} \tag{8}$$

where μ is the centroid position of tactile depth images, and Σ is the pixel summation of tactile depth images which represents the intensity of contact.

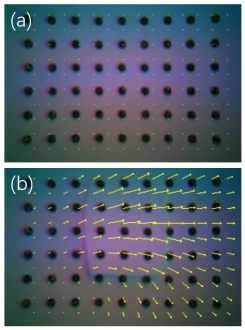

Fig. 2. Optical flow when the object is stationary (a) has a tendency to move to the right (b). The yellow arrow indicates the vector of optical flow for the sampled points. The entropy of optical flow distributions is considered as an estimation of shear.

For tactile dynamic features, slide friction will alter the shape of the gel on the surface of the tactile sensor. Therefore, they can be captured by the field displacement of the gel. In this study, we employ the optical flow method and use the Lucas-Kanade method to calculate it, as shown in Fig. 2. Finally, we get the optical flow field as follows:

$$F = \begin{bmatrix} \{P(x_1, y_1), P(x_2, y_2), \cdots P(x_n, y_n)\} \\ \{Q(x_1, y_1), Q(x_2, y_2), \cdots Q(x_n, y_n)\} \end{bmatrix} \tag{9}$$

Here, F represents the optical flow field, where P and Q are the component functions of the optical flow vector. F gives optical flow vector at every (x_i, y_i) point, which could be considered as a collection of samples derived from the optical flow distribution. It will provide some useful features. For instance, we calculate the distribution entropy of F in two directions based on the distribution of their values, which can feature the magnitude of shear forces [21]. The higher the optical flow entropy, the greater the motion trends of the object, resulting in larger deformations of the gel. This process can be formalized as follows:

$$\begin{aligned} H(x) &= -\sum p(P) \log p(P) \\ H(y) &= -\sum p(Q) \log p(Q) \end{aligned} \tag{10}$$

where H denotes the distribution's entropy, and $p(P)$ and $p(Q)$ are the discrete distributions estimated by samples histogram. It is worth noting that a larger H indicates a more dispersed distribution of the optical flow, signifying a greater trend of motion.

Thus, the tactile information o_t of our framework contain: μ, Σ, $H(x)$ and $H(y)$. The information above encompasses both dynamic and static characteristics of the object at the time of contact, such as contact location, contact intensity, and slipping trend. The selection of these features is deemed sufficient for many tasks.

IV. EXPERIMENTS

A. Task and Experiment Setup

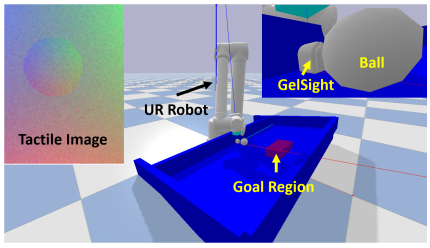

Fig. 3. Object pushing: The UR5 manipulator is tasked with pushing a ball or box from the lower of the slope to the upper red goal region.

To validate our algorithm and compare it with others, we focus on robot manipulation tasks, which are ideal for applying artificial intelligence due to the challenges of decision planning in dynamic, unstructured environments. Using a UR robot equipped with tactile information, we benchmarked various approaches in simulation by having the robot push a ball or box on a slope with dense or sparse rewards, evaluating learning efficiency and adaptability. The simulation was built using PyBullet and the TACTO tactile sensor plugin [22], which offers fast processing of RGB and depth images. The scene in which the UR robot pushes an object on a slope is depicted in Figure 3.

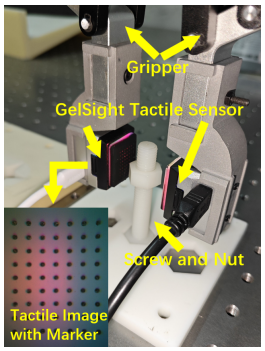

Fig. 4. Robot screwing: The manipulator equipped with the gripper will adaptively hold the nylon nut and utilize tactile sensor feedback to screw it.

In our physical robot experiments, we focus on screwing nuts using tactile feedback, a common industrial scenario illustrated in Figure 4. The primary challenge is the unknown pitch parameter of the screw, making the robot must explore through touch. Improper strategies may cause shear between tactile feedback and the screw nut, prompting the need for reinforcement learning to develop more effective twisting strategies despite motion deviations. This experiment was conducted using the KUKA iiwa7 manipulator, the BackYard gripper, and the GelSight mini tactile sensor.

B. Environment Implementation Details

A key element of reinforcement learning is the interaction of dataflow. In this section, we provide a detailed description of our dataflow configuration, which is formalized using the Gym wrapper.

The observation of the state vector, denoted as o_p, in the simulation consists of two parts: the position, velocity, posture, and angular velocity of the manipulator and the object to be manipulated. So the action a involves three degrees of freedom for incremental motion: moving forward, left/right, and rotation. And for physical robots, we have limited the movements to only two: descent and rotation around the descent direction. Therefore, in this case, o_p includes only the descent height and rotation angle. Here, we enable the end effector to move downward at a fixed speed, while the action a is rotation incremental motion.

Tactile features are represented as o_t, which can be tailored according to the characteristics of the task. For example, in the simulation environment, it is impossible to obtain the optical flow of tactile sensors, hence o_t only includes the μ and Σ of the contact depth images, providing a comprehensive description for pushing objects.

Reward is usually the key determinant of the success of a reinforcement learning algorithm. In the simulation task, we have devised two reward configurations. The sparse reward configuration assigns a reward of 1.0 when the ball or box reaches the goal region, while the dense-reward configuration additionally incorporates the negative distance between the object and the center of the goal region. For physical robots, our objective is to achieve an appropriate angular speed for the rotation of the screw nut and minimize the shear force exerted on the screw nut in the vertical direction. Unfortunately, obtaining the ground-truth angle of the nut is challenging. Therefore, we employ the entropy of optical flow as a suitable substitute. The reward is formulated as the negative entropy in the downward direction, aiming to minimize deformation in this particular direction. The reward function mentioned above can be formalized as shown in Equation 11, where r_1, r_2, and r_3, respectively, represent the dense reward function in the simulation, the sparse reward function in the simulation, and the screwing reward function in the experiments. Here, $\text{sgn}(get\_target)$ is used to determine whether the object is within the target area, returning a value of 0 or 1, and $\text{dis}(object, target)$ calculates the distance from the pushed object to the center of the target area.

$$\begin{aligned} r_1 &= \text{sgn}(get\_target) - \text{dis}(object, target) \\ r_2 &= \text{sgn}(get\_target) \\ r_3 &= -H(y) \end{aligned} \tag{11}$$

C. Result and Discussion

In the simulation, we compared our method with the vanilla RL method, Soft Actor-Critic (SAC) and our approach performs well. Additionally, we ran our method three times using different random seeds to plot the mean and variance. However, when it came to physical robots, we only implemented our method due to limitations posed by sampling costs. To mitigate data fluctuations, we computed rewards based on a sliding window of 10 episodes for simulations and 5 episodes for experiments.

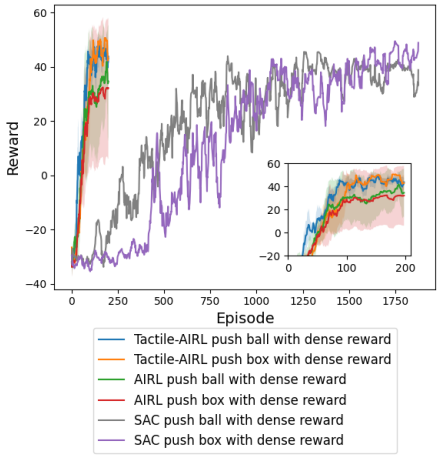

Fig. 5. Experiments in dense reward configuration. Compared with our method, origin AIRL mothod and SAC for pushing ball and pushing box.

1) Simulation with Dense Reward and Sparse Reward: In dense reward settings, our method, along with most reinforcement learning approaches, can learn the skill of pushing. As shown in the comparison results in Figure 5, we found that Tactile-AIRL and origin AIRL are both an order of magnitude more efficient than the RL baseline. They achieved nearly the maximum cumulative reward after only 100 episodes of interaction with the environment, whereas SAC required approximately 1000 episodes. The reason behind this improvement is our adoption of a modeling approach similar to model-based reinforcement learning, thus significantly improving data efficiency.

Another noteworthy point is the role of tactile sensors in the process of learning manipulation skills. We found that the absence of tactile sensors makes the learning of manipulation skills more unstable, leading to larger variances and relatively lower smooth rewards in the result curves. This is because tactile sensors provide more local information, which is challenging to obtain from o_p.

In the sparse reward configuration, most RL methods cannot learn the skill of pushing. As depicted in Figure 6, we observed that the RL baseline failed to learn even after exploring thousands of episodes, whereas our method consistently achieved high performance. This capability arises from the algorithm's objective of minimizing free energy, which drives it to acquire new information during exploration until it discovers states with effective rewards.

2) Physical Experiment: We train physical experiment until the reward of the recent 5 episodes reaches a relatively stable level, achieving the skill of minimizing downward shear force, as depicted in Figure 7. Our approach demonstrates convergence in 15 episodes, providing strong evidence for the data efficiency of our algorithm in real-world applications. Incorporation of tactile sensation enables us to

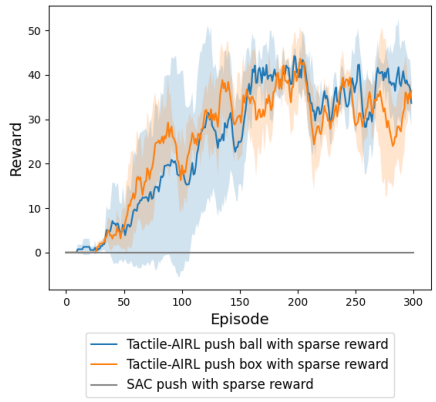

Fig. 6. Experiments in Sparse Reward Configuration. In this case, it has been observed that SAC fails to learn skills.

estimate the effectiveness of task execution and enhances the perception space. Moreover, the use of active inference reinforcement learning expedites skill acquisition, highlighting the potential of our algorithm for broader application scenarios.

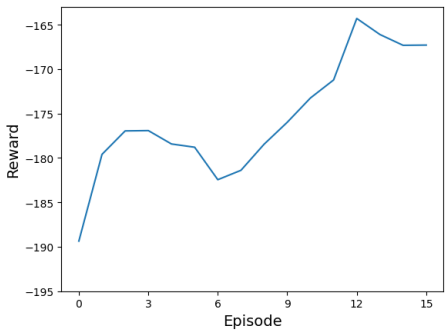

Fig. 7. Experiments in physical robot. As the value of the reward increases, the shear force during the screwing process is reduced.

V. CONCLUSIONS

We propose an efficient learning method for robot manipulation skills, called Tactile-AIRL, which uses Active Inference RL to train robot agents in environments equipped with tactile sensing. The method encourages agents to explore effectively and utilizes the tactile features of exploration interactions for model learning, synchronously improving both exploration and exploitation. Experiments show that Tactile-AIRL has unique advantages in the field of robotics.

REFERENCES

[1] A. Delgado, C. Jara, and F. Torres, "In-hand recognition and manipulation of elastic objects using a servo-tactile control strategy," *Robotics and Computer-Integrated Manufacturing*, vol. 48, pp. 102–112, 2017.

[2] N. Jamali, C. Ciliberto, L. Rosasco, and L. Natale, "Active perception: Building objects' models using tactile exploration," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, 2016, pp. 179–185.

[3] R. Li, R. Platt, W. Yuan, A. ten Pas, N. Roscup, M. A. Srinivasan, and E. Adelson, "Localization and manipulation of small parts using gelsight tactile sensing," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 3988–3993.

[4] D. Hafner, T. Lillicrap, J. Ba, and M. Norouzi, "Dream to control: Learning behaviors by latent imagination," *arXiv preprint arXiv:1912.01603*, 2019.

[5] A. Yamaguchi and C. G. Atkeson, "Combining finger vision and optical tactile sensing: Reducing and handling errors while cutting vegetables," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, 2016, pp. 1045–1051.

[6] N. Kuppuswamy, A. Alspach, A. Uttamchandani, S. Creasey, T. Ikeda, and R. Tedrake, "Soft-bubble grippers for robust and perceptive manipulation," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 9917–9924.

[7] Y. She, S. Wang, S. Dong, N. Sunil, A. Rodriguez, and E. Adelson, "Cable manipulation with a tactile-reactive gripper," *The International Journal of Robotics Research*, vol. 40, no. 12-14, pp. 1385–1401, 2021.

[8] S. Wang, J. Wu, X. Sun, W. Yuan, W. T. Freeman, J. B. Tenenbaum, and E. H. Adelson, "3d shape perception from monocular vision, touch, and shape priors," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 1606–1613.

[9] C. Wang, S. Wang, B. Romero, F. Veiga, and E. Adelson, "Swingbot: Learning physical features from in-hand tactile exploration for dynamic swing-up manipulation," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 5633–5640.

[10] E. Smith, R. Calandra, A. Romero, G. Gkioxari, D. Meger, J. Malik, and M. Drozdzal, "3d shape reconstruction from vision and touch," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 14 193–14 206.

[11] J. Wu, G. Xin, C. Qi, and Y. Xue, "Learning robust and agile legged locomotion using adversarial motion priors," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 4975–4982, 2023.

[12] X. Liu, G. Wang, Z. Liu, Y. Liu, Z. Liu, and P. Huang, "Hierarchical reinforcement learning integrating with human knowledge for practical robot skill learning in complex multi-stage manipulation," *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 3, pp. 3852–3862, 2024.

[13] A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, and T. Funkhouser, "Learning synergies between pushing and grasping with self-supervised deep reinforcement learning," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4238–4245.

[14] Y. Luo, Y. Wang, K. Dong, Q. Zhang, E. Cheng, Z. Sun, and B. Song, "Relay hindsight experience replay: Self-guided continual reinforcement learning for sequential object manipulation tasks with sparse rewards," *Neurocomputing*, vol. 557, p. 126620, 2023.

[15] F. R. Karl Friston, Thomas FitzGerald, P. Schwartenbeck, J. O'Doherty, and G. Pezzulo, "Active inference and learning," *Neuroscience & Biobehavioral Reviews*, vol. 68, pp. 862–879, 2016.

[16] K. Friston, "The free-energy principle: a unified brain theory?" *Nature Reviews Neuroscience*, vol. 11, no. 2, pp. 127–138, Feb 2010.

[17] C. Pezzato, R. Ferrari, and C. H. Corbato, "A novel adaptive controller for robot manipulators based on active inference," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2973–2980, 2020.

[18] A. Tschantz, B. Millidge, A. K. Seth, and C. L. Buckley, "Reinforcement learning through active inference," *arXiv preprint arXiv:2002.12636*, 2020.

[19] T. Schneider, B. Belousov, G. Chalvatzaki, D. Romeres, D. K. Jha, and J. Peters, "Active exploration for robotic manipulation," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 9355–9362.

[20] S. Wang, Y. She, B. Romero, and E. Adelson, "Gelsight wedge: Measuring high-resolution 3d contact geometry with a compact robot finger," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 6468–6475.

[21] W. Yuan, R. Li, M. A. Srinivasan, and E. H. Adelson, "Measurement of shear and slip with a gelsight tactile sensor," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 304–311.

[22] S. Wang, M. Lambeta, P.-W. Chou, and R. Calandra, "TACTO: A fast, flexible, and open-source simulator for high-resolution vision-based tactile sensors," *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, no. 2, pp. 3930–3937, 2022.