# dVRL: Reinforcement Learning Environments for Surgical Robotics

Florian Richter[1] *Student Member, IEEE*, Ryan K. Orosco[2] *Member, IEEE*,
and Michael C. Yip[1] *Member, IEEE*

***Abstract*— Reinforcement Learning (RL) is a framework that recently has found success by integrating Artificial Intelligence to solve a variety of complex problems. We aim to bring the successes from the RL community to the surgical robotics community by presenting the first open-sourced RL environemnts for surgical robotics, dVRL[3]. By engaging the broader community, which includes both surgical robotics and non-domain experts such as reinforcement learning enthusiasts, new solutions can be contributed to problems that would have real world significance to robotic surgery and the patients that undergo those procedures. To show the effectiveness of the simulated environments, learned policies are transferred to the real robot and sucessfully accomplish surgically relevant tasks.**

## I. Introduction

Reinforcement Learning (RL) is a framework that integrates Artificial Intelligence to solve variety of complex problems [1]. The framework is based on a Markov Decision Process where a control policy is learned through interaction with the environment and maximizes the long term reward, which is specified appropriately to solve the problem. This allows for model-free controllers to be found that can solve challenging, non-linear problems. Alongside the ongoing evolution and successes of RL, there is active research in automating surgical tasks [2]. One of the challenges moving forward for the surgical robotics community is that a lot of the recent work is based on hand-crafted control policies that can be difficult to both develop at scale and generalize well.

We aim to bridge these two communities by presenting dVRL, the first open-sourced RL environments for Surgical Robotics. We are motivated to engage the broader community that includes surgical robotics and also non-domain experts, such that reinforcement learning enthusiasts with no domain knowledge of surgery can still easily prototype their algorithms with such an environment and contribute to solutions that would have real world significance to robotic surgery and the patients that undergo those procedures. To evaluate performance of the presented environments, learned policies are transferred to the real robot with minimal effort to automate a surgically relevant task.

## II. Methods

The environments are simulated in V-REP based on the da Vinci® Surgical System System scenes developed by Fontanelli et al. [3]. The presented environments only incorporate a single Patient Side Manipulator (PSM), as shown in Fig. 1, but the additional PSM and endoscopic camera arm can be easily added. The PSM arm is controlled in the end-effector space and, for the sake of simplicity, the orientation is held constant. By working in the end-effector space, the learned policies can be transferred to various EndoWrists which are the attachable surgical tools for the da Vinci® Surgical System and have varied kinematic chains. For the simulated scene in the environments presented, the Large Needle Driver (LND) EndoWrist is used, but it can be replaced with other EndoWrist models.

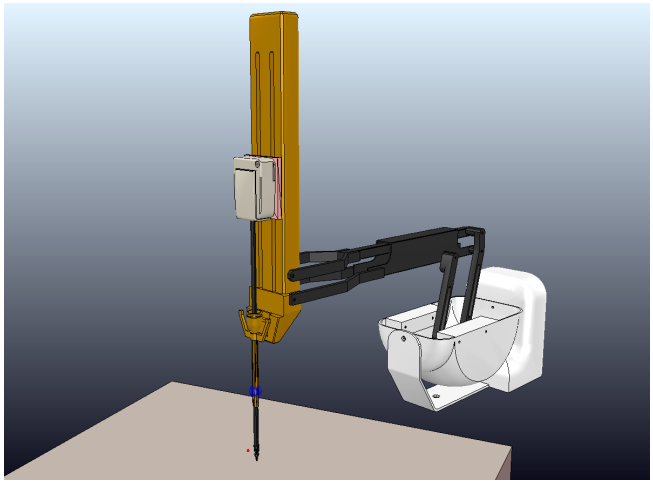

Fig. 1: Simulation scene in V-REP of the single PSM arm. This is the fundamental scene that the presented environments, PSM Reach and PSM Pick, are based on.

### A. PSM Reach Environment

The goal of the PSM Reach Environment is to have the end-effector reach a target goal position $g$ starting from any initial configuration, $p_0$. Note that both $g$ and $p_0$ are 3-dimensional vectors describing end-effector position and when resetting the environment are randomly sampled from the workspace of the PSM arm. The state and action space of the environment is:

$$s_t = \begin{bmatrix} p_t & g \end{bmatrix} \qquad a_t = \begin{bmatrix} \Delta_t \end{bmatrix} \tag{1}$$

where $\Delta_t$ is the control input that changes the end-effector position, which is bounded by [-1, 1]. The next state equation for the position is:

$$p_{t+1} = \eta \Delta_t + p_t \tag{2}$$

where $\eta$ is a scaling factor. The term $\eta$ must be kept low enough to ensure stability on the real da Vinci® Surgical System. This is since every new action gives new set points to the joint level controllers. If the new set points are far from the current position, overshoot and instability can occur. The reward function is:

$$r(s_t) = -\mathbb{1}_{||p_t - g|| > \delta} \tag{3}$$

where $\mathbb{1}$ is the indicator function and $\delta$ is a threshold distance.

[1]Florian Richter and Michael C. Yip are with the Department of Electrical and Computer Engineering, University of California San Diego, La Jolla, CA 92093 USA. {frichter, yip}@ucsd.edu

[2]Ryan K. Orosco is with the Department of Surgery - Division of Head and Neck Surgery, University of California San Diego, La Jolla, CA 92093 USA. rorosco@ucsd.edu

[3]dVRL available at https://github.com/ucsdarclab/dVRL

### B. PSM Pick Environment

The goal of the PSM Pick Environment is to bring an object, with 3D position $o_t$, to a goal position $g$. In this case, the object is a small simulated cylinder. The PSM grippers jaw angle, $j_t$, is also activated and bounded from 0 to 1, similar to LND on the real da Vinci® Surgical System. When resetting the environment, $g$ is randomly sampled from the workspace, $p_0$ is set to be above the object, and $o_0$ is set to a constant position on the table shown in Fig. 1. The state and action space of the environment is:

$$s_t = \begin{bmatrix} p_t & j_t & o_t & g \end{bmatrix} \qquad a_t = \begin{bmatrix} \Delta_t & j_{t+1} \end{bmatrix} \tag{4}$$

where $p_t$ is defined as previously stated and the action sets both a change in position and the jaw angle directly. The reward function for the environment is:

$$r(s_t) = -\mathbb{1}_{||o_t - g|| > \delta} \tag{5}$$

where $\delta$ is once again the threshold distance.

## III. Experiments

Both PSM Reach and Pick environments have $\delta$ set to 3mm and $\eta$ set to 1mm. This value for $\eta$ was found to be the highest value where no overshoot was observed through experimentation on the da Vinci Research Kit (dVRK) [4]. The environments were trained using Deep Deterministic Policy Gradient with Hindsight Experience Replay (DDPG + HER) [5]. Due to sparsity of the PSM Pick environment, the loss function for the actor in DDPG is augmented with a behavioral cloning loss from demonstrations [6].

The learned policies were transferred to the real da Vinci® Surgical System using dVRK [4]. The surgical task the policies are tested on is to use the suction and irrigation EndoWrist tool to remove fake blood from a simulated abdomen to reveal shrapnel. Then the shrapnel must be removed using the other PSM with the LND EndoWrist. The suction and irrigation tool uses the learned PSM Reach Policy, and the LND tool uses a composition of the learned PSM Reach and PSM Pick policies. The positional information and jaw angles are measured and set utilizing the encoder readings, forward and inverse kinematics, and joint level controllers implemented in dVRK [4]. The goals are preset by manually moving the arms to the goal locations and recording the positional information.

## IV. Results

The PSM Reach environment successfully trained a learned policy using DDPG + HER where the goal is reached within $\delta$ distance 100% of the time. The PSM Pick environment did not successfully train a learned policy within 30000 training episodes using only DDPG + HER, but with the behavioral cloning, a learned policy successfully brought the object within $\delta$ distance 100% of the time. Fig. 2 shows the learned policy transfer experiment and the completion of the surgical task. Both the PSM Reach and Pick policies reach the goal within $\delta$ distance 100% of the time.

## V. Discussion and Conclusion

In this work, we present the first, open-sourced reinforcement learning environment for surgical robotics called dVRL. The learned policies effectively transferred to the real world and solved surgically relevant tasks. We see dVRL as enabling the broad surgical robotics community to fully leverage the newest strategies in reinforcement learning, and for reinforcement learning scientists with no previous domain knowledge of surgical robotics to be able to test and develop new algorithms that can have real-world, positive impact to patient care and the future of autonomous surgery.

## References


[1] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[2] M. Yip and N. Das, *ROBOT AUTONOMY FOR SURGERY*, ch. Chapter 10, pp. 281–313.

[3] G. A. Fontanelli *et al.*, "A v-rep simulator for the da vinci research kit robotic platform," in *BioRob*, 2018.

[4] P. Kazanzides *et al.*, "An open-source research kit for the da vinci ®surgical system," *IEEE Intl. Conf. on Robotics and Automation*, pp. 6434–6439, 2014.

[5] M. Andrychowicz *et al.*, "Hindsight experience replay," in *Advances in Neural Information Processing Systems*, 2017.

[6] A. Nair *et al.*, "Overcoming exploration in reinforcement learning with demonstrations," in *IEEE Intl. Conf. on Robotics and Automation*, pp. 6292–6299, 2018.


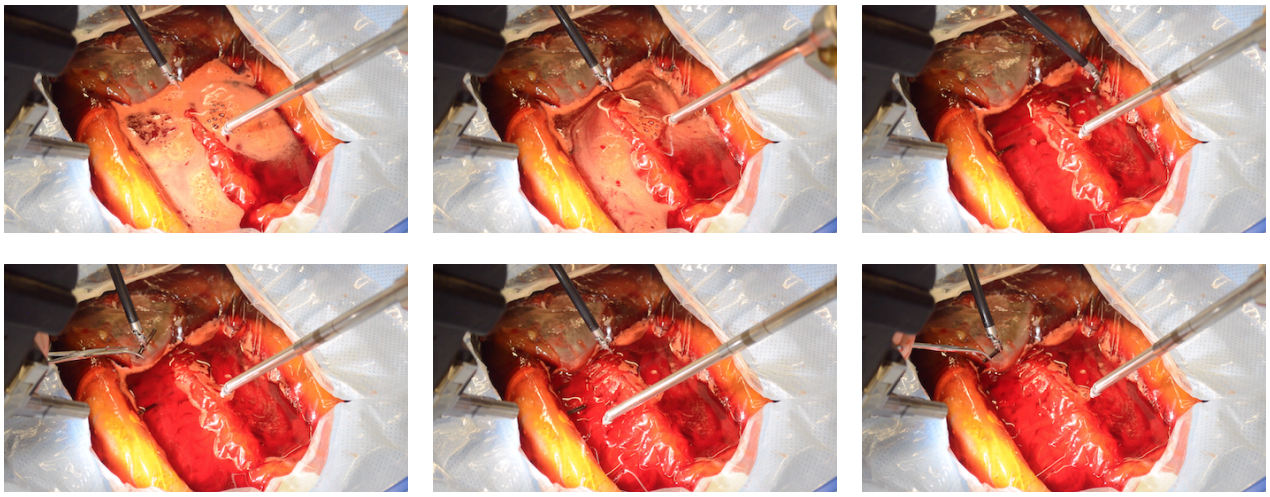

Fig. 2: The suction tool using a learned PSM Reach policy to remove fake blood to reveal debris. After the debris is revealed, the Large Needle Driver utilized a composition of learned PSM Reach and PSM Pick policies to remove the debris and hand it to the first assistant.