Tobias Behr*, Tim Philipp Pusch, Marius Siegfarth, Dominik Hüsener, Tobias Mörschel and Lennart Karstensen

Deep Reinforcement Learning for the Navigation of Neurovascular Catheters

<https://doi.org/10.1515/cdbme-2019-0002>

Abstract: Endovascular catheters are necessary for state-of-the-art treatments of life-threatening and time-critical diseases like strokes and heart attacks. Navigating them through the vascular tree is a highly challenging task. We present our preliminary results for the autonomous control of a guidewire through a vessel phantom with the help of Deep Reinforcement Learning. We trained Deep-Q-Network (DQN) and Deep Deterministic Policy Gradient (DDPG) agents on a simulated vessel phantom and evaluated the training performance. We also investigated the effect of the two enhancements Hindsight Experience Replay (HER) and Human Demonstration (HD) on the training speed of our agents. The results show that the agents are capable of learning to navigate a guidewire from a random start point in the vessel phantom to a random goal. This is achieved with an average success rate of 86.5% for DQN and 89.6% for DDPG. The use of HER and HD significantly increases the training speed. The results are promising and future research should address more complex vessel phantoms and the use of a combination of guidewire and catheter.

Keywords: Deep Reinforcement Learning, Catheter, Guidewire, Endovascular, DQN, DDPG, Hindsight Experience Replay, Human Demonstration, Stroke, Heart Attack, Neurovascular

1 Introduction

Endovascular catheters are necessary for state-of-the-art treatments of life-threatening and time-critical diseases like strokes and heart attacks. Navigating a set of catheter and guidewire (henceforth simply referred to as catheter) through the vascular tree is a highly challenging task. The dynamics of

Figure 1 Closed loop for autonomous passive catheter navigation

catheters are difficult to predict or calculate and vascular trees have a high variance in shape. Therefore movements at the proximal end of the catheter translate into unexpected movements at the distal tip due to friction and elastic deformation of catheter and vessel wall. For this reason, conventional control approaches are hardly applicable for catheter navigation. Robotic assistance, to date, is limited to telemanipulation or making the catheter controllable by having a steerable or magnetic tip [1].

In order to overcome the current limitations of catheter control, we propose a closed loop control system based on Deep Reinforcement Learning (DRL) as shown in Figure 1 for the autonomous navigation of standard passive catheters. The system, once fully developed, will use images from the angiography system to track the catheter position in real time. This position data will be fed into a control algorithm which was trained for the task of navigating catheters. The control algorithm will provide a robotic catheter manipulator with

*Corresponding author: Tobias Behr: Fraunhofer IPA, Mannheim, Germany, e-mail: tobias.behr@ipa.fraunhofer.de
 Tim Philipp Pusch, Marius Siegfarth, Dominik Hüsener, Tobias Mörschel and Lennart Karstensen: Fraunhofer IPA, Mannheim, Germany

control commands which the manipulator will use to steer the catheter through the vasculature.

The angiography system is already the state of the art in modern catheter labs and robot manipulators are the focus of several research groups. With R-One by Robocath or CorPath by Corindus a few catheter manipulators are already commercially available. However, both catheter tracking and control algorithms remain to be developed and are the focus of our joint research project in cooperation with Fraunhofer MEVIS. While Fraunhofer MEVIS focuses on the tracking algorithm we focus on the development of the control algorithm using DRL. This paper aims to discuss the feasibility of this approach by presenting the first results for a DRL based control system using a passive guidewire in an idealized vascular phantom.

2 Materials and Methods

The experimental setup consists of test bench, catheter tracking, control algorithm and a catheter simulation for the generation of training data. Component interaction is implemented using OpenIGTLink [2].

2.1 Control Algorithm

The inputs of the control algorithm are the absolute coordinates of the guidewire tip as well as the target and the rotation of the manipulator. Outputs are either continuous for each degree of freedom (DOF) (rotation and translation) or discrete with eight different movement options. The discrete options represent every possible combination of the two DOF excluding the “no-movement-at-all” option.

The training of the control algorithm is achieved using the Deep Reinforcement Learning (DRL) approach [3]. With this approach an agent interacts with an environment and learns how to improve its interactions by receiving rewards for its actions as shown in Figure 2. At the beginning of the training process the agent is completely unaware of its task and of the environment. Its first actions therefore are performed randomly. The rewards the agent receives from the environment are used to optimize the neural net. By updating each neural net parameters with respect to its influence on the quality of the actions the agent learns how to perform good actions and ceases to perform bad ones. By and by the actions improve and originate more and more from experience instead of randomness.

Table 1 lists the different kinds of rewards an agent can achieve throughout an episode of training. In our setup reward per

Figure 2 Training loop for an agent using a neural net trained with Deep Reinforcement Learning

interaction step is either sparse which means the agent simply receives -1 point for every step and an additional +1 when the goal is reached or dense with -0.02 generally, an additional +1 when the target is reached and +0.02 multiplied with the distance from the tip of the catheter to the goal for every step. The distance is calculated as the normalized distance along the centreline.

For continuous control, we use Deep Deterministic Policy Gradients (DDPG) [3] partially enhanced by Hindsight Experience Replay (HER) [4] and Human Demonstration (HD) [5]. Discrete control is achieved by a Deep Q-Network (DQN) [6] with the extension of double Q-learning, dueling networks and prioritized experience replay [7]. The implementations by OpenAI Baselines [8] and RLlib [9] are used respectively. The neural networks consist of three hidden layer of 256 neurons for DDPG and two hidden layers of 128 neurons for DQN.

Table 1 Calculation of rewards. Total rewards are calculated as the sum of the rewards of the individual steps.

Sparse Reward	-1	for every interaction step
	+1	once if the goal is reached
Dense Reward	-0.02	for every interaction step
	+0.02	multiplied with the distance of catheter tip and goal, for every interaction step
	+1	once if the goal is reached

2.2 Simulation Environment

The movement of the guidewire through the vascular tree is simulated using SOFA [10] with the beam adapter plugin [11]. The walls of the vascular tree are rigid and the friction coefficient between wall and guidewire is 0.1. The lumen is empty, thus no dynamic resistance to the guidewire motion is considered. The simulation routine consists of receiving the velocities for both DOFs, calculating the movement for the appropriate number of steps and sending the catheter position to the control algorithm.

2.3 Catheter Tracking

In the current experimental setup, the guidewire is monitored by a camera mounted above the vessel phantom. The images are processed in real-time by a processing pipeline in MeVisLab, MeVis Medical Solutions AG, Bremen. To extract the catheter shape, the camera image is registered to a CT image of the phantom to mask the non-vessel parts of the image. In the vessel part, the guidewire is extracted via threshold segmentation, its centreline is extracted, and sent to the control algorithm.

2.4 Test Bench

Figure 3 Vessel phantom made of PMMA with a guidewire inserted

At this stage, we are evaluating the feasibility of DRL for catheter control using only one component of the catheter set, namely the guidewire. Our test bench therefore features an experimental robotic manipulator, which can handle the two DOF of a guidewire. The maximum linear travel is 300 mm while the rotation is infinite. The first idealized vessel phantom, as shown in Figure 3, is made of PMMA with a constant vessel diameter of 10mm and it has a bifurcation followed by a bi- and trifurcation in one plane. Starting points

and targets are chosen randomly. A camera is mounted above for tracking. The phantom is filled with water for better visibility and less wall friction.

2.5 Experiments

The aim of our experiments was to evaluate, whether the use of DRL may be considered a promising approach for the autonomous navigation of passive catheters. We used the simulation environment described in 2.2 to train the neural nets and recorded their training performances. We trained them with the help of two DRL agents and evaluated their performance in an idealized vessel phantom as a first proof-of-concept. These agents were Deep-Q-Networks (DQN) for discrete control commands and Deep Deterministic Policy Gradient (DDPG) for continuous ones. We performed two sets of experiments. The first set was carried out to find possible advantages of one DRL agent over the other. The second set of experiments was carried out to evaluate if the use of the enhancements Hindsight Experience Replay and Human Demonstration improve the training speed of the agents. Only DDPG is used on the test bench and the second experiment set, because continuous control is preferred to discrete control and the first set will show little difference in the success rates.

3 Results

Figure 4 Success rate of the agents over training episodes

Figure 4 shows that both agents are capable of learning to navigate a guidewire from a random start point in the vessel phantom to a random goal. This is achieved with an average success rate of 89.6% for DDPG + HER using sparse reward and 86.5% for DQN + Rainbow using dense reward. Training is considered finished when the success rate is higher than the average of all the following success rates for the first time. The success rate is measured every 1000 episodes. DDPG finishes

training after about 11000 episodes and DQN after about 5000 episodes. The standard deviation for four training attempts is 4.4% for DDPG and 6.2% for DQN. Both agents use standard hyperparameters as given by the OpenAI Baselines and RLlib. After transferring the trained DDPG model to the test bench, we reached the goal 7 out of 10 times. The failed attempts happened at junctions, where exact state data is necessary to be able to navigate reliably. The simulation does not yet include noise and latency.

Figure 5 Comparison of the training speeds of plain Deep Deterministic Policy Gradient (DDPG), DDPG + Hindsight Experience Replay (HER) and DDPG + HER + Human Demonstration (HD)

Figure 5 shows a comparison of the training speeds of a DDPG model with and without the use of Hindsight Experience Replay (HER) and Human Demonstration (HD). Training for plain DDPG finished after 12.3 hours (15000 episodes) and achieved a success rate of 83.1%. Training of DDPG with HER and HER + HD both finish after about 9.0 hours (11000 episodes) and achieved success rates of 89.6% and 87.9% respectively. The use of DDPG + HER + HD led to a significantly faster increase of the success rate during the first hour of training than when using just DDPG or DDPG+HER. Training 1000 episodes took 0.82h on a computer equipped with the CPU AMD Ryzen 7 2700X, GPU GeForce RTX 2070 and 48 GB RAM.

4 Discussion

Both DQN and DDPG perform quite similar in an idealized vessel phantom. DQN trains faster while DDPG achieves more stable results and requires less domain specific knowledge for reward calculation. The results are overall promising given that no case-specific adaptations have yet been made to the baseline agent algorithms. The comparison of training speeds

of different enhancements of DDPG suggests that the training speed can be increased significantly using Hindsight Experience Replay and Human Demonstration. Fast training is especially interesting for the future for which we plan to have a pre-trained agent learn on patient-specific vasculature prior to an intervention. Today the image data of stroke patients often arrive at the Stroke Unit before the patient does because the patient was first diagnosed in a more remote hospital. This data could be used to train the pre-trained agent on this patient's specific anatomy while waiting for him or her to arrive at the Stroke Unit.

To date, we have been using highly idealized vascular phantoms and just a guidewire to investigate the feasibility of our approach. The results show that baseline agents for DRL are able to achieve promising results in these idealized phantoms using similar informational content as exists in real interventions. Future research should address the limitations mentioned by adapting the agents and the setup to more realistic vascular geometries and the use of both guidewire and catheter.

Author Statement

This work was supported by the Fraunhofer Internal Programs under Grant No. WISA 833 967. The authors state no conflict of interest.

References

- [1] H. Rafii-Tari, C. J. Payne, G.-Z. Yang, Current and emerging robot-assisted endovascular catheterization technologies: A review, *Ann. Biomed. Eng.*, vol. 42, no. 4, pp. 697-715, 2014.
- [2] J. Tokuda, et. al. OpenIGTLink: An open network protocol for image-guided therapy environment: *Int. J. Med. Robot. Comput. Assist. Surg.*, 2009, 5, (4), pp. 423-434 (doi: 10.1002/rcs.274)
- [3] T. Lillicrap et al., Continuous Control with Deep Reinforcement Learning, *arXiv:1509.02971v2*, 2015
- [4] M. Andrychowicz, et al. Hindsight Experience Replay: *Advances in Neural Information Processing Systems*, pages 5048-5058, 2017.
- [5] Nair et al, Overcoming Exploration in Reinforcement Learning with Demonstrations, *arXiv:1709.10089v2*, 2018
- [6] V. Mnih et al., Playing Atari with Deep Reinforcement Learning, *arXiv:1312.5092*, 2013
- [7] M. Hessel, et al., Rainbow: Combining Improvements in Deep Reinforcement Learning. *arXiv:1710.02298*
- [8] OpenAI Baselines, <https://github.com/openai/baselines>
- [9] RLlib, <https://ray.readthedocs.io/en/latest/rllib.html>
- [10] F. Faure, et al., "SOFA: A Multi-Model Framework for Interactive Physical Simulation", *Soft tissue biomechanical modeling for computer assisted surgery*. Springer, Berlin, Heidelberg, 2012. 283-321.
- [11] C. Duriez, et al., "New approaches to catheter navigation for interventional radiology simulation", *Computer aided surgery* 11, no. 6, 2006: 300-3