IEEE Robotics and Automation Letters (RAL) paper presented at
2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

FNPG-NH: A Reinforcement Learning Framework for Flexible Needle Path Generation with Nonholonomic Constraints

Mukund Shah, Niravkumar Patel *Member, IEEE*

***Abstract*—Path planning algorithms for minimally invasive neurosurgery involve avoiding critical structures such as blood vessels and ventricles while following needle kinematics. The majority of planning solutions proposed in the literature use sampling-based algorithms. This paper introduces a Flexible Needle Path Generation framework with Non-Holonomic constraints (FNPG-NH), an extension of our FNPG framework. FNPG-NH uses deep Reinforcement Learning (RL) based methods such as Deep Deterministic Policy Gradient (DDPG), Proximal Policy Optimization (PPO), and Soft Actor-Critic (SAC) to obtain a kinematically feasible path for a bevel-tipped flexible needle using a nonholonomic model. RL algorithms presented in this work generate the control input for needle rotation based on the rewards generated by the environment. The deep RL algorithms are trained on an environment that consists of (1) ventricles segmented from T1 images of the healthy volunteers using atlas-based segmentation, (2) blood vessels segmented from MRA volumes of the same volunteer using thresholding, and (3) tumor volume from labeled BraTS 2020 dataset and placed at an anatomically relevant location. The paths generated by the reinforcement learning algorithm and the traditional sampling-based algorithm RRT are compared for various performance metrics. The reinforcement learning model was trained on 20 volumes and validated on 68 volumes, and RRT was evaluated on the same 68 validation volumes. The results show that the trajectories generated by the FNPG-NH framework are safer, shorter, and take less time than RRT while avoiding critical structures such as ventricles and blood vessels.**

***Index Terms*—Surgical Robotics: Planning, Steerable Catheters/Needles; Reinforcement Learning**

I. INTRODUCTION

NEUROSURGERY is a medical process that involves diagnosing and treating disorders related to the central nervous system. Historically, brain surgeries have been performed via craniotomy, which causes long-lasting side effects. For the last few decades, various neurosurgery interventions have been performed using minimally invasive methods, such as keyhole-based neurosurgical procedures.

Conventionally minimally invasive neurosurgery procedures are performed using rigid needles, which limit access to desired lesions while avoiding critical structures. Therefore steerable catheters [1] and flexible needles [2] are now widely used for minimally invasive surgeries as they provide higher dexterity, safety, and stability. However, trajectory planning for a steerable needle in a minimally invasive surgery is a challenging and time-consuming task as not only the generated path has to avoid the obstacles and reach the goal, but it also needs to follow the kinematic constraints of the steerable needle. Traditionally, the neurosurgeon uses three orthogonal views of the MRI (Magnetic Resonance Imaging) volume to plan a safe trajectory for straight, rigid needles; however, manually planning a safe trajectory for a flexible bevel-tipped needle is not feasible as it follows a curved path, and can steer in any direction by changing the orientation of the tip of the needle. To solve this trajectory planning problem, we present a Flexible Needle Path Generation framework with Non-Holonomic constraints (FNPG-NH) that uses deep RL to autonomously generate safe trajectories for minimally invasive neurosurgery procedures using flexible bevel-tipped needles.

The autonomous surgical planning framework FNPG-NH presented here uses (1) atlas-based segmentation of brain MRI-T1 volumes to identify critical structures such as ventricles, (2) thresholding-based segmentation of brain MR angiography (MRA) volumes from the IXI dataset [3], and (3) labeled tumors from the BraTS 2020 dataset [4]–[6] to create a simulated environment. It uses a deep RL-based algorithm to generate a safe trajectory while enforcing the constraints of a nonholonomic bevel-tipped needle [7]. Our proposed method of solving the planning problem with deep RL is trained on simulated environments that are anatomically relevant. The FNPG-NH framework can generate a safe path for the steerable flexible needle from an autonomously generated skull-entry point and a target region while avoiding the critical anatomical structures. This framework could be used for preoperative planning for minimally invasive neurosurgery procedures performed using flexible bevel-tipped needles.

II. RELATED WORKS

Conventionally minimally invasive procedures are performed using rigid needles. Finding a safe path using rigid needles is challenging, as they cannot be steered away from critical anatomical structures. In recent years, using flexible, steerable needles for performing these surgical procedures made it possible to reach targets that were impossible for rigid needles to reach. Neurosurgeons are required to plan a needle path that starts from the skull entry point (burr hole) and reaches the desired target (tumor) while avoiding

Manuscript received: April, 15, 2023; Revised June, 06, 2023; Accepted July, 18, 2023.

This paper was recommended for publication by Editor Pietro Valdastri upon evaluation of the Associate Editor and Reviewers' comments.

Mukund Shah and Niravkumar Patel are with the Department of Engineering Design at Indian Institute of Technology Madras, Chennai 6000036, India mukundshah1729@gmail.com, nirav.robotics@gmail.com

Digital Object Identifier (DOI): see top of this page.

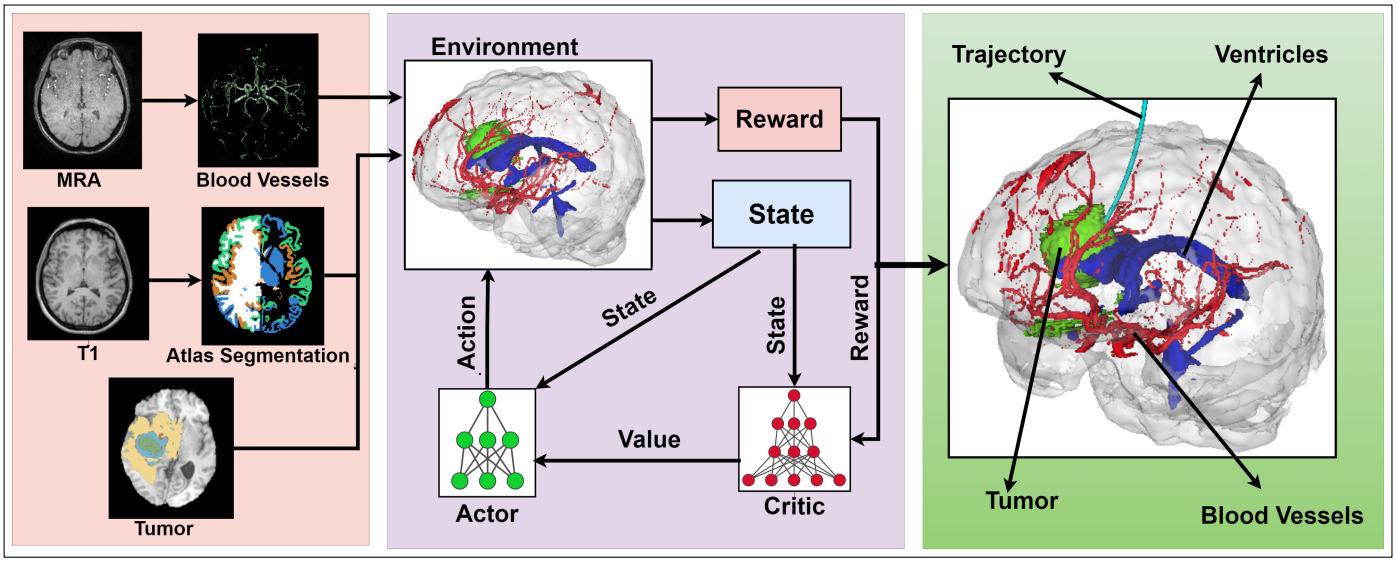

Fig. 1: Block diagram of the FNPG-NH framework showing the modules for data preparation, actor-critic deep RL algorithms, and resultant safe needle trajectory overlaid on the segmented MRI volume.

the critical anatomical structures. This path planning problem for steerable flexible needles becomes more complex than the rigid ones, as the needle path not only has to avoid the critical anatomical structures but must also follow the kinematic constraints of the needle. Many researchers have explored various algorithmic approaches over the last few years to address this path-planning problem for flexible, steerable needles, including grid search algorithms, sampling-based motion planning algorithms, and, more recently, reinforcement learning-based solutions.

Park et al. presented a diffusion-based planning method for a flexible needle under kinematic constraints in an obstacle-free space [8]. Duindam et al. presented a 3D path planning algorithm for steerable bevel-tipped needles that considered the kinematic constraints of the needle for planning in the presence of obstacles; they discretized the control space of the needle and solved it as a non-linear convex optimization problem. However, this method did not guarantee global convergence [9].

Khatib et al. proposed potential field methods that find paths from start to goal configuration by minimizing the potential function in the C-space [10]. DiMaio et al. demonstrated using the potential fields with tissue deformation and needle deflection model to plan a path [11]. However, the potential field method does not guarantee convergence; hence it may not find a solution even if one exists. Graph-based search methods discretize the planning problem. Djisktra [12] and A* [13] are two of the most popular graph search algorithms, and given a planning problem, they find the optimal path. But as the search space dimensions increase, the computational time for these algorithms also increases, making them less desirable for higher dimensional planning problems. Sampling-based algorithms randomly sample a point from the configuration space and reduce the computation time compared to graph-based search methods; however, the sampling-based algorithms are probabilistically complete and may not find a solution in a given time, even if one exists. RRT [14] and its improved variants RRT* [15] and bidirectional RRT [16] are popular algorithms for performing planning in higher dimensional spaces.LaValle et al. showed using RRT with kinematic constraints for planning in higher dimensions [16]. Ayoung et al. and Yongkang et al. demonstrated the use of RRT [17] and RRT* [18] in a 3D environment following the kinematic constraints of the flexible needle.

Deep reinforcement learning algorithms have recently been explored in path planning and have shown promising results compared to sampling-based and graph-based algorithms. Panov et al. successfully applied a neural Q-learning algorithm to solve a path planning problem in a small grid world environment [19]. Xiaoyun et al. demonstrated the use of deep RL in dynamic environments using lidar sensor information and showed the generalization behavior of reinforcement learning algorithms [20]. Junli et al. combined off-policy deep RL algorithm TD3 with probabilistic roadmaps and showed that it outperforms traditional graph search algorithms like A* in both small and large-scale planning problems [21]. In the context of minimally invasive neurosurgery, Alice et al. used asynchronous advantage actor-critic algorithm, which combines Deep Q-Network (DQN) with actor-critic algorithms, learns policy by steps proportional to the advantage function, and performs asynchronous updates [22]; furthermore, they demonstrated the use of inverse deep RL-based framework for intraoperative path planning with a replanning time of 0.02 s [23]. In our prior work, we used the deep RL-based framework FNPG for generating a path for the flexible, steerable needle, however, without the nonholonomic needle constraints [24].

Our previous work [24] showed the use of deep RL for needle trajectory planning. However, that work did not incorporate the nonholonomic kinematic constraints of the needle and did not use anatomically relevant tumor data. This work presents FNPG-NH, a flexible needle path planning framework

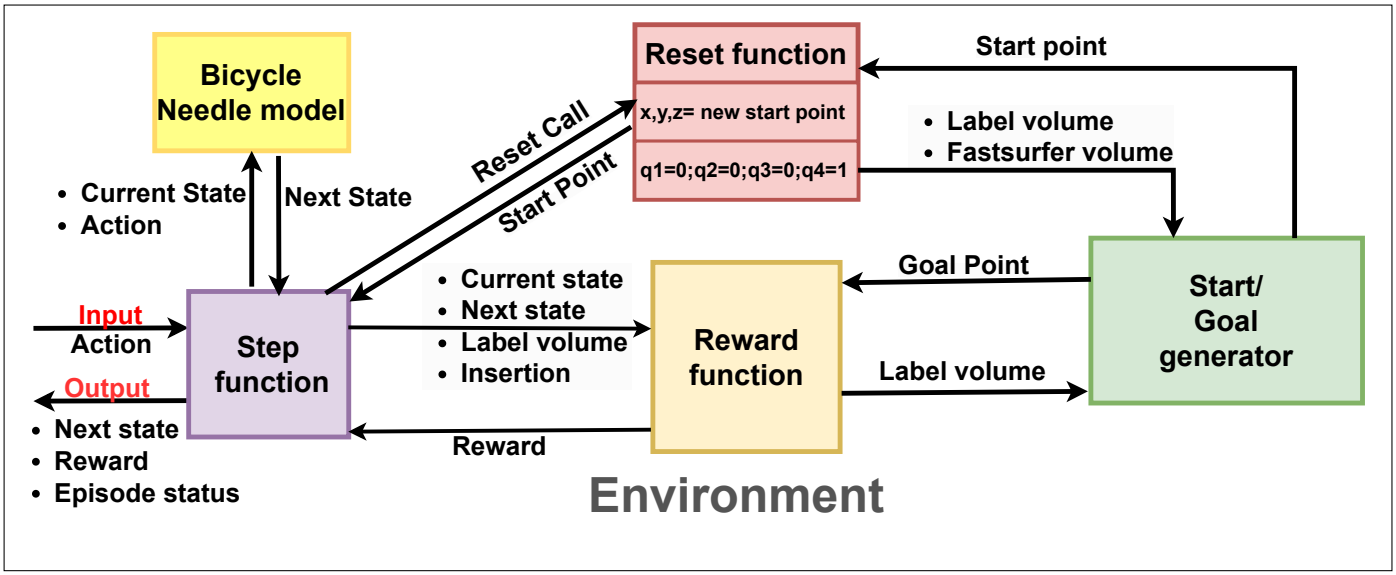

Fig. 2: Block diagram the simulated RL environment: 1) input to the environment is an action as angle of needle rotation in *rad*, 2) output of the environment is the next state, reward, and episode status, 3) states in the diagram are seven-dimensional, with x,y, and z representing the position and q1,q2,q3, and q4 are the quaternions representing the orientation of the needle tip.

for steerable, bevel-tipped needles with nonholonomic constraints. FNPG-NH is based on deep RL algorithms, namely DDPG [25], PPO [26], and SAC [27] to generate a trajectory for a bevel-tipped needle while enforcing the nonholonomic kinematic constraints.

Major contributions of the presented work are (1) the development and evaluation of an RL-based FNPG-NH framework for autonomous preoperative path planning of a steerable bevel-tipped needle for minimally invasive neurosurgery procedures, (2) to the best of our knowledge, this is the first work where the deep RL framework was trained and evaluated with a needle model having nonholonomic kinematic constraints, (3) presented framework does not require starting point from the surgeon, it autonomously finds one for an optimal trajectory, (4) presented deep RL framework was evaluated in an anatomically relevant dataset prepared by merging (a) MRI-T1 and MRA of healthy volunteers and (b) labeled tumor (excluding edema) from BraTS 2020 dataset and (5) the extensive (68 volumes) evaluation of the presented framework shows that the FNPG-NH framework produces safer and shorter needle trajectories in less time compared to the sampling-based algorithm.

III. Method

In this section, the modules of the FNPG-NH framework: (A) data preparation and (B) trajectory generation using deep reinforcement learning, are explained. Figure 1 shows a block diagram of the presented FNPG-NH framework.

A. Data Preparation

The datasets used for environment creation are IXI dataset with MRI-T1 and MRA volumes of the same volunteers and BraTS 2020 dataset with segmentation label map of the tumor of the patients.

*1) **Atlas-based Segmentation**:* FastSurfer neuroimaging pipeline [28] is used to create a segmentation map of the brain. The brain MRI-T1 image is first preprocessed by FastSurfer and conformed to a size of 256 x 256 x 256 with an isotropic voxel size of 1 mm and image intensities scaled between 0-255. FastSurfer consists of 3 F-CNNs operating on coronal, axial, and sagittal 2D slices and a final view aggregation stage. Each FCNN comprises four competitive dense block encoder decoder layers followed by a final competitive dense block layer. In this work, an MRI-T1 brain volume from the IXI dataset is provided as input to the pre-trained FastSurfer network, and from the obtained segmentation map, only ventricles are used as critical structures and the brain stem is used as a reference for choosing the region for needle entry points, this is further explained in Sec III-B4.

*2) **Blood Vessel Segmentation**:* The brain MRA volumes were first conformed to the size of 256 x 256 x 256 with a voxel size of 1 mm to match the dimensions of the MRI-T1 volumes and voxel intensities were scaled between 0-255. To eliminate the difference in the translation part of the affine matrix of the T1 and MRA volumes, an affine correction was done to ensure that the origin of both volumes was at the same location. Thresholding was done on the affine-corrected MRA volumes. The threshold was kept at 67 percent of maximum intensity, and voxels with intensities greater than 67 percent were labeled as blood vessels.

*3) **Tumor placement**:* The IXI dataset used for segmentation of the ventricles and blood vessels is from healthy volunteers; hence it does not contain tumors. To augment an anatomically relevant tumor model, a tumor label volume was randomly selected from the available training set of the BraTS 2020 dataset. Since the tumor comes from a different dataset, following tumor placement steps were followed to ensure it gets embedded at an anatomically relevant position

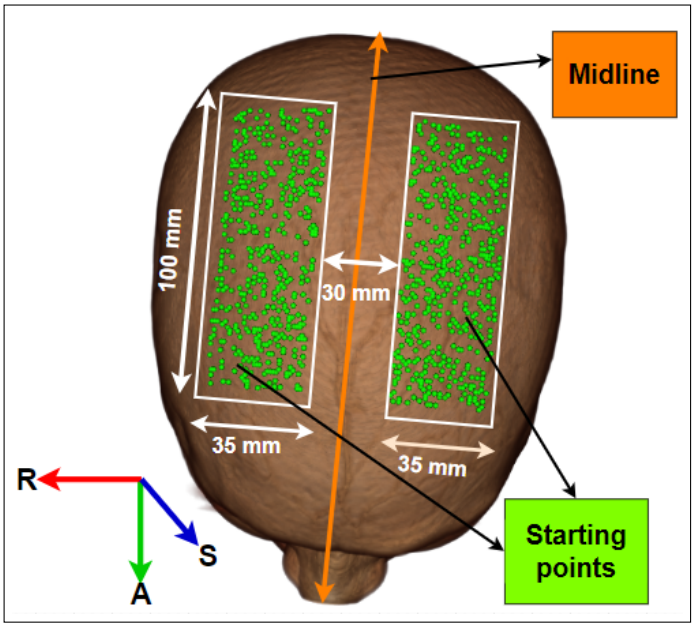

Fig. 3: Shows the valid start point selection region created by 1000 random starting points. We leave a space of 15 mm on the left and right of the midline, and the starting points are selected from a region of 100 mm x 35 mm.

with respect to the segmented IXI dataset.

1) Conform the tumor label to match the dimensions of brain MRA and MRI-T1 volumes, i.e., 256 x 256 x 256 with a voxel size of 1 mm.
2) Get the enhancing tumor label from the BraTS 2020 dataset.
3) Place the enhancing tumor volume such that the tumor's inferior most point lies above the brainstem's centroid.
4) Convert the IXI MRI-T1 volume into a binary label map to remove the background (thresholding with intensities > 0).
5) Perform logical *AND* operation between the binary IXI MRI-T1 and tumor label volume.
6) Measure the overlap (number of voxels) between the binary IXI MRI-T1 and tumor label volume.
7) Consider the present tumor volume as anatomically relevant if the measured overlap was > 90 percent of the original tumor volume.

The above criteria ensure that the tumor is placed randomly in an anatomically relevant location (i.e tumor is inside the skull and not below the brain stem). After a valid tumor volume is found, the whole solid tumor consisting of the enhancing, non-enhancing, and necrotic/cystic labels from the corresponding volume was used as a tumor volume. Here, the brain stem is extracted from the segmented label map of the IXI MRI-T1 volume input to the FastSurfer pipeline.

B. Path Generation

1) ***Model:*** Our path generation module uses actor-critic deep RL algorithms for continuous actions, DDPG, PPO, and SAC. The implementations used are from an open-source library, stablebaselines3 [29].

DDPG: The actor and critic both have two fully connected layers of size [400,300]. The activation function used for the first layer of both networks is *ReLU*. The final layer uses *Tanh* activation for the actor and a linear activation for the critic. A replay buffer of size 10^6 is used and a learning rate of 10^{-3} is used for both networks. A mini-batch size of 100 with a gamma (discount factor) value of 0.99 and a tau (Polyak update) value of 0.005 is used. For exploration, Ou noise [30] was used. Target actor and critic network architectures were cloned from the actor and critic architectures respectively and initialized with the same weights as the actor and the critic networks.

PPO: The actor and critic both have two fully connected layers of size [64,64]. *ReLu* activation is used in the first layer of both actor and critic networks. *Tanh* is used as the activation for the second layer of the actor and a linear activation is used for the critic. Mini-batch of size 64, a learning rate of 3 x 10^{-4}, a gamma value of 0.99, and a generalized advantage estimate (gae) lambda value of 0.95 and a clip range of 0.2 were used.

SAC: Two fully connected layers of size [256,256] were used as actor and critic networks. The activation function used for the first layer is *ReLU* for both networks. The final layer activations are *Tanh* and linear for the actor and the critic networks, respectively. A replay buffer of size 10^6 was used, a mini-batch of size 256 along with a learning rate of 3 x 10^{-4}, gamma (discount factor) value of 0.99, and tau (soft update coefficient) value of 0.005 were used for training.

Actor-critic algorithms have two sets of parameters. The critic network updates the action value function Q(s, a) by temporal difference policy evaluation. The actor network updates policy parameters in the direction suggested by the critic network by using the policy gradient theorem. The above models use an environment implemented using OpenAI gym [31](version 0.21.0), and well-established hyperparameters mentioned above.

2) ***Environment:*** The environment is a segmented volume that consists of three voxel labels: safe, unsafe, and tumor. The unsafe voxels consist of the ventricles taken from the segmentation output of the MRI-T1 volume and the blood vessels taken from the segmentation of the MRA volume. The tumor label is taken from the segmented T1 volume of the BraTS 2020 dataset. The safe voxels are those that are not labeled as unsafe or tumor. The environment consists of the *step* and *reset* functions. The *step* function takes action as input and returns the next state, reward, and episode status. The *reset* function calls the *start-point selection* function Sec III-B4 and restarts the algorithm from a new randomly selected valid start point. Figure 2 shows the block diagram of the training environment.

3) ***Reward Function:*** The *reward* function takes input from a segmented volume consisting of critical structures (ventricles and blood vessels) treated as obstacles and goal (tumor), current state, next state, goal point (centroid of the tumor), maximum insertion depth, and total insertion as the inputs. The reward has five components:

(1) Distance reward: We use a Euclidean distance-based reward function which gives a positive reward of +1 if the

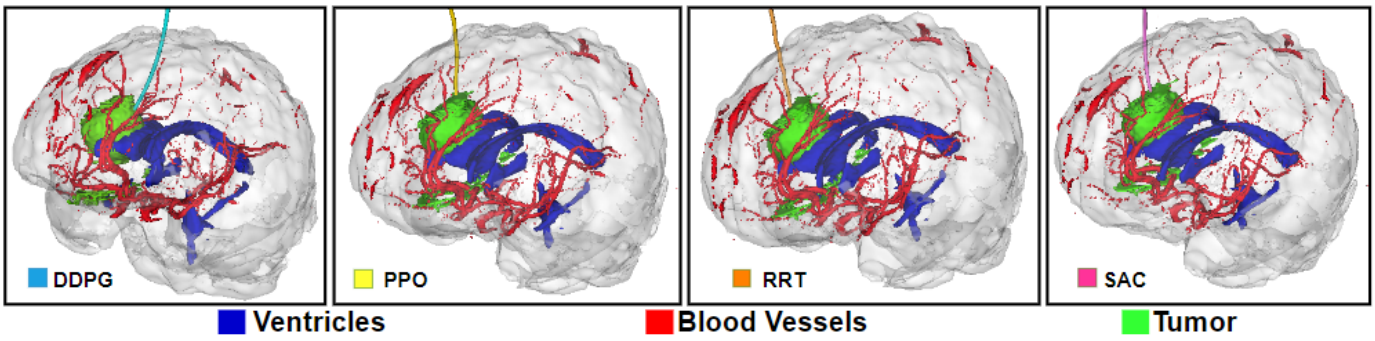

Fig. 4: Showing the environment consisting of segmented blood vessels (red), ventricles (blue), and tumor (green) with first: trajectory generated by DDPG (light blue), second: trajectory produced by PPO (yellow), third: trajectory produced by RRT (orange) and forth: trajectory produced by SAC (purple) for different entry points

needle tip moves closer to the goal and a negative reward of -1 if the needle tip moves away from the goal.

(2) Out-of-bounds reward: If the agent reaches outside the bounds of the volume, it receives a penalty of -1.

(3) Obstacle reward: If the agent hits an obstacle, it receives a high penalty of -1000.

(4) Tumor reward: A high positive reward of +1000 is awarded if the agent hits the tumor.

(5) Target Reward: A high positive reward of +10000 is given when the needle tip reaches the target region that is defined as a sphere or radius 5 mm from the centroid of the tumor.

The final reward comes out to be the sum of all five components. The intuition behind the Euclidean distance-based reward function is to guide the model to make the needle tip reach the tumor. The reset flag is set to *True* when the total insertion is either equal to or exceeds the maximum insertion depth of 150 mm.

*4) **Start-point selection**:* The start point selection region shown in Fig. 3 is considered based on the inputs from a collaborating neurosurgeon. After every call to the *reset* function of the environment, a new star point is selected from a region, as shown in Fig. 3. The start point selection region is defined using the following procedure: 1) The centroid of the brainstem from the atlas-based segmentation of the MRI-T1 volume is used as a reference. 2) The centroid of the brainstem is projected towards the superior direction to find the midline on the skull. 3) The region of 15 mm on the left and the right side of the mid-line is left out. 4) Points are randomly selected on a region of 35 mm x 100 mm on either the left or right side of the midline, as shown in Fig. 3.

IV. Experiments and Results

Data preparation, model training, and testing were performed using an Ubuntu 20.04.4 LTS machine with an AMD Ryzen 5, 6 Core CPU with 16 GB RAM.

A. Data

Data was prepared with two freely available datasets: IXI and BraTS 2020. IXI dataset contains MRI-T1 and MRA volumes of healthy volunteers. For the sake of completeness of the framework, we used tumors from the BraTS 2020 dataset and placed them randomly at anatomically relevant places as discussed in Sec. III-A3 inside the brain. This data preparation strategy is explained in Sec. III-A.

B. Training

The position and orientation of the needle tip is defined by a SE^3 state space. The first three components of the state are the x,y, and z in the R^3 space signifying the position of the needle tip. The next four components are quaternions (q1,q2,q3, and q4) in the SO^3 space signifying the orientation of the needle tip. As shown in Fig. 2, the training involves calling the *step* and *reset* functions. The *reset* function calls the *start point generation* function, which selects a random starting point from an anatomically relevant region shown in Fig. 3. The quaternions are always set to values 0,0,0, and 1 every time *reset* is called. This helps the algorithm to learn a policy that generalizes over the anatomy of the brain. As mentioned in Sec. III-B2 and shown in Fig. 2, the *step* function takes action as an argument and returns the next state, reward, and episode completion status. Here the action is one-dimensional and controls the angular input to the nonholonomic bicycle needle model. In every step, the needle is inserted by a fixed length of 5 mm at a fixed curvature of 0.01 mm $^{-1}$ while following the nonholonomic kinematic constraints. For faster convergence, the *reward function* constantly forces the agent to move toward the tumor.

Results are based on a total of 88 labeled volumes. We used 20 volumes to train the RL algorithms and the remaining 68 volumes to test both RRT and RL algorithms. Figure 4 shows the trajectories generated by DDPG, PPO, RRT, and SAC in the simulated environment. The RL algorithms were tested with 68 volumes and ten random start points per volume, resulting in a total of 680 trajectories. Each trajectory was validated by a nonholonomic model-based motion validator to ensure that the trajectory does not interact with the critical structures (obstacles). The FNPG-NH framework with the deep RL models DDPG, PPO, and SAC was compared with sampling-based kinodynamic RRT with the same starting points that were generated by the RL algorithms.

The following metrics were used to compare the above models:

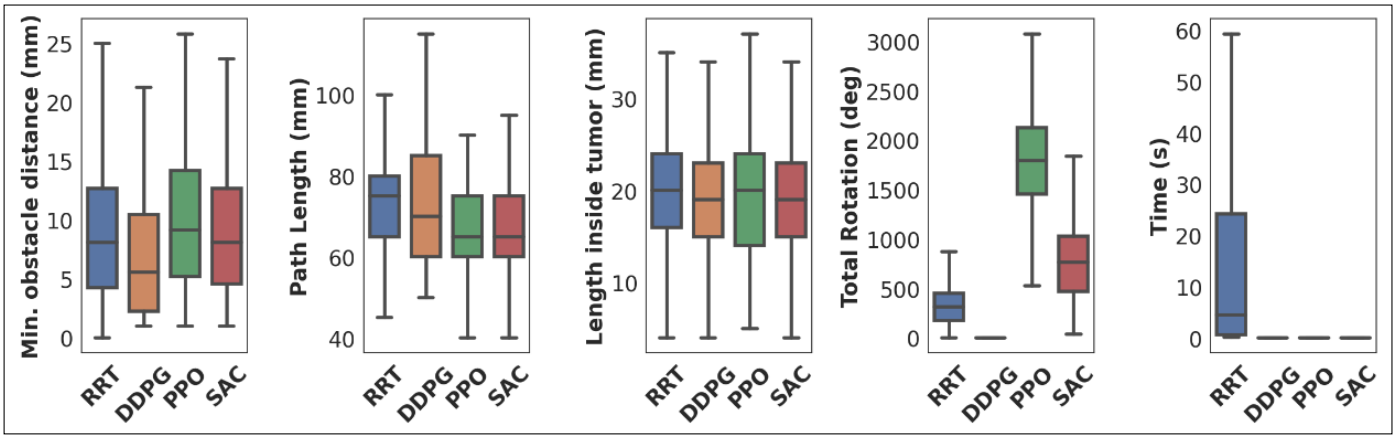

Fig. 5: Comparison of minimum obstacle distance, path length, length of trajectory inside the tumor, total rotation, and computation time between RRT, DDPG, PPO, and SAC-based models.

- **Path length:** Path length signifies the total insertion depth from a valid starting point to the goal for a volume.
- **Computation time:** Computation time measures the time taken by the algorithm to generate a trajectory from a start point to the goal.
- **Distance from critical structures:** Ventricles and blood vessels are the critical structures in our environment, and the minimum distance from them signifies the safety of the trajectory generated by the algorithm.
- **Total rotation:** This metric signifies the total amount of rotation done by corresponding control inputs at each point in the path. It is calculated by accumulating the absolute difference between the successive control inputs.
- **Length inside tumor:** This metric signifies the needle length inside the tumor. Usually, longer insertion depth inside the tumor provides a better biopsy sample.

The comparison between all the algorithms based on the metrics above is shown in Fig. 5. Observed results were statistically analyzed on the above-mentioned metrics.

Bartlett's variance test was performed to check for equality of variance among the group of algorithms for each metric. It was observed that minimum obstacle distance ($p > 0.05$) and length inside the tumor ($p > 0.05$) showed equal variances. Whereas, path length ($p < 0.05$), computation time ($p < 0.05$), and total rotation ($p < 0.05$) showed unequal variances. Results were evaluated for statistical significance using a one-way ANOVA test and Welch's ANOVA test for the metrics with equal and unequal variances, respectively; all the metrics were statistically significantly different ($p < 0.05$). To get individual pairwise comparisons among the algorithms on the aforementioned metrics, post hoc tests, namely Tukey's Honestly significant difference test and pairwise Games-Howell test were performed.

Tables 1-5 summarize the results.Table I shows the minimum distance of the needle trajectory from the critical structures (the larger this distance, the better it is). Tukey's Honestly significant post hoc analysis was used for this metric. It was observed that PPO (9.93 mm) outperformed DDPG (6.80 mm), RRT (8.82 mm), and SAC (9.09 mm) with a statistically significant difference ($p < 0.05$). DDPG performed the worst as compared to PPO, RRT, and SAC ($p < 0.05$). No statistically significant difference was observed between RRT & SAC ($p > 0.05$).

TABLE I: Summary of Minimum Obstacle Distance shown for different quantiles

	Minimum obstacle distance (mm)				
	25%	50%	75%	mean	min
RRT	4.24	8.12	12.68	8.82	0.00
DDPG	2.23	5.56	10.48	6.80	1.00
SAC	4.58	8.12	12.65	9.09	1.00
PPO	5.19	9.16	14.23	**9.93**	1.00

TABLE II: Summary of Path Length shown for different quantiles

	Path length (mm)				
	25%	50%	75%	mean	min
RRT	65.00	75.00	80.00	73.81	45.00
DDPG	60.00	70.00	85.00	73.00	50.00
SAC	60.00	65.00	75.00	**66.41**	40.00
PPO	60.00	65.00	75.00	66.44	40.00

Table II shows the needle path length for different algorithms. Post hoc analysis using the Games-Howell test shows that the mean path length (the lower the length, the better it is) for SAC (66.41 mm) and PPO (66.44 mm) was statistically significantly ($p < 0.05$) better than RRT (73.81 mm) and DDPG (73.00 mm). However, no statistically significant difference was found among DDPG & RRT ($p > 0.05$) and PPO & SAC ($p > 0.05$).

Table III shows the length of the path that went inside the tumor (the larger it is, the better it is). Tukey's Honestly significant post hoc analysis of mean length inside the tumor showed that: RRT (19.97 mm) performed statistically significantly better ($p < 0.05$) than SAC (18.72 mm). No significant difference was found among DDPG (19.27 mm) & RRT (19.97 mm), PPO (19.09 mm) & RRT (19.97 mm), SAC (18.72 mm) & DDPG (19.27 mm), and SAC (18.72 mm) & PPO (19.09

TABLE III: Summary of Length of trajectory inside tumor shown for different quantiles

	Length inside tumor (mm)				
	25%	50%	75%	mean	min
RRT	16.00	20.00	24.00	**19.97**	2.00
DDPG	15.00	19.00	23.00	19.27	4.00
SAC	15.00	19.00	23.00	18.72	4.00
PPO	14.00	20.00	24.00	19.09	5.00

TABLE IV: Summary of Total Rotation shown for different quantiles

	Total rotation (deg)				
	25%	50%	75%	mean	min
RRT	172.5	313.1	453.6	350.4	0.0
DDPG	0.0	0.0	0.0	**0.0**	0.0
SAC	469.6	767.5	1025.9	755.9	36.9
PPO	1456.9	1791.5	2124.0	1811.6	527.0

TABLE V: Summary of Computation Time shown for different quantiles

	Computation time (s)				
	25%	50%	75%	mean	min
RRT	0.582	4.471	24.146	15.960	0.131
DDPG	0.027	0.035	0.045	**0.037**	0.017
SAC	0.034	0.040	0.046	0.040	0.022
PPO	0.032	0.038	0.045	0.039	0.019

mm) with p-values > 0.05.

Table IV presents the total rotation metric (the lesser it is, the better it is). The Games-Howell post hoc analysis showed the following order for the mean total rotation (from best to worst): DDPG (0 deg) $>$ RRT (350.4 deg) $>$ SAC (755.9 deg) $>$ PPO (1811.6 deg), with all of them being statistically significantly different ($p < 0.05$).

Table V presents the computation time taken by each algorithm (the lower the time, the better it is). Games-Howell post hoc analysis of mean computation time shows that DDPG (0.037 s), PPO (0.039 s), and SAC (0.040 s) outperformed RRT (15.960 s) with statistically significantly lower computation time ($p < 0.05$). Also, DDPG outperformed SAC with a statistically significant difference ($p < 0.05$). No statistically significant difference was found between DDPG & PPO ($p < 0.05$) and PPO & SAC ($p < 0.05$).

V. Discussions and Conclusions

In this paper, we introduced a reinforcement learning-based framework FNPG-NH for non-holonomic needle steering problems. It was observed that our framework produces shorter (PPO and SAC) and safer (PPO) trajectories in a relatively short amount of time (DDPG, SAC, and PPO) as compared to the sampling-based algorithm (RRT). Also, we found that DDPG produced the trajectories with the least amount of total rotation; however, this led to a lower success rate for DDPG than PPO and SAC. In our opinion, this behavior of DDPG can be attributed to the learned policy favoring a certain action to maximize the rewards; DDPG tries to find a starting point from where a rotation-free path exists toward the goal while avoiding obstacles. Unlike DDPG, SAC has an entropy of policy in the objective, giving more spread action choices. We can further improve the results of the off-policy actor-critic algorithms DDPG and SAC by using Hindsight Experience Replay (HER) [32]; HER gives the ability to learn from the failures of an agent. Learning-based algorithms were also observed to have a higher success rate than sampling-based RRT. Also, RL frameworks could generalize over the anatomy of the patients with less amount of data and could further improve if more variety of training data is provided. Based on the statistical analysis of the presented results, we can conclude that PPO performed the best among all algorithms in most of the metrics and produced safer, shorter trajectories quickly.

In the future, we plan to improve the following aspects of the FBPG-NH framework: (1) update the start point selection routine such that they are selected on the same side of the location of the tumor to generate trajectories that are surgically relevant, (2) variable curvature based on tissue and needle mechanical properties, and (3) explore other reward functions that could include more needle manipulation criteria such as variable insertion step size, total needle rotation and allow angular needle insertions from the starting point, and (4) perform a thorough hyperparameter search for the Deep RL algorithms.

References

[1] X. Hu, A. Chen, Y. Luo, C. Zhang, and E. Zhang, "Steerable catheters for minimally invasive surgery: a review and future directions," *Comput Assist Surg (Abingdon)*, vol. 23, pp. 21–41, Dec. 2018.

[2] L. Frasson, S. Y. Ko, A. Turner, T. Parittotokkaporn, J. F. Vincent, and F. Rodriguez y Baena, "STING: a soft-tissue intervention and neuro-surgical guide to access deep brain lesions through curved trajectories," *Proc Inst Mech Eng H*, vol. 224, no. 6, pp. 775–788, 2010.

[3] L. Imperial College London, South Kensington Campus, "Ixi dataset." http://brain-development.org/ixi-dataset/.

[4] B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, *et al.*, "The multimodal brain tumor image segmentation benchmark (BRATS)," *IEEE Trans Med Imaging*, vol. 34, pp. 1993–2024, Dec. 2014.

[5] S. Bakas, H. Akbari, A. Sotiras, M. Bilello, M. Rozycki, J. S. Kirby, J. B. Freymann, K. Farahani, and C. Davatzikos, "Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features," *Sci Data*, vol. 4, p. 170117, Sept. 2017.

[6] S. Bakas, M. Reyes, A. Jakab, S. Bauer, *et al.*, "Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge," 2018.

[7] R. J. Webster, J. S. Kim, N. J. Cowan, G. S. Chirikjian, and A. M. Okamura, "Nonholonomic modeling of needle steering," *The International Journal of Robotics Research*, vol. 25, pp. 509 – 525, 2006.

[8] W. Park, J. S. Kim, Y. Zhou, N. Cowan, A. Okamura, and G. Chirikjian, "Diffusion-based motion planning for a nonholonomic flexible needle model," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pp. 4600–4605, 2005.

[9] V. Duindam, R. Alterovitz, S. S. Sastry, and K. Goldberg, "Screw-based motion planning for bevel-tip flexible needles in 3d environments with obstacles," *2008 IEEE International Conference on Robotics and Automation*, pp. 2483–2488, 2008.

[10] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, vol. 2, pp. 500–505, 1985.

[11] S. P. DiMaio and S. E. Salcudean, "Needle steering and motion planning in soft tissues," *IEEE Trans Biomed Eng*, vol. 52, pp. 965–974, June 2005.

[12] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, pp. 269–271, Dec 1959.

[13] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.

[14] S. M. LaValle, "Rapidly-exploring random trees : a new tool for path planning," *The annual research report*, 1998.

[15] M. Jordan and A. Perez, "Optimal bidirectional rapidly-exploring random trees," 2013.

[16] S. M. LaValle and J. James J. Kuffner, "Randomized kinodynamic planning," *The International Journal of Robotics Research*, vol. 20, no. 5, pp. 378–400, 2001.

[17] A. Hong, Q. Boehler, R. Moser, A. Zemmar, L. Stieglitz, and B. J. Nelson, "3D path planning for flexible needle steering in neurosurgery," *Int J Med Robot*, vol. 15, p. e1998, May 2019.

[18] Y. Zhang, Z. Qi, and H. Zhang, "An improved rrt* algorithm combining motion constraint and artificial potential field for robot-assisted flexible needle insertion in 3d environment," in *2021 3rd International Conference on Industrial Artificial Intelligence (IAI)*, pp. 1–6, 2021.

[19] A. I. Panov, K. S. Yakovlev, and R. Suvorov, "Grid path planning with deep reinforcement learning: Preliminary results," *Procedia Computer Science*, vol. 123, pp. 347–353, 2018. 8th Annual International Conference on Biologically Inspired Cognitive Architectures, BICA 2017 (Eighth Annual Meeting of the BICA Society), held August 1-6, 2017 in Moscow, Russia.

[20] X. Lei, Z. Zhang, and P. Dong, "Dynamic path planning of unknown environment based on deep reinforcement learning," *Journal of Robotics*, vol. 2018, p. 5781591, Sep 2018.

[21] J. Gao, W. Ye, J. Guo, and Z. Li, "Deep reinforcement learning for indoor mobile robot path planning," *Sensors*, vol. 20, no. 19, 2020.

[22] A. Segato, L. Sestini, A. Castellano, and E. De Momi, "Ga3c reinforcement learning for surgical steerable catheter path planning," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2429–2435, 2020.

[23] A. Segato, M. Di Marzo, S. Zucchelli, S. Galvan, R. Secoli, and E. De Momi, "Inverse reinforcement learning intra-operative path planning for steerable needle," *IEEE Transactions on Biomedical Engineering*, vol. 69, no. 6, pp. 1995–2005, 2021.

[24] J. Kumar, C. S. Raut, and N. Patel, "Automated flexible needle trajectory planning for keyhole neurosurgery using reinforcement learning," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4018–4023, 2022.

[25] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.

[26] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *ArXiv*, vol. abs/1707.06347, 2017.

[27] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018* (J. G. Dy and A. Krause, eds.), vol. 80 of *Proceedings of Machine Learning Research*, pp. 1856–1865, PMLR, 2018.

[28] L. Henschel, S. Conjeti, S. Estrada, K. Diers, B. Fischl, and M. Reuter, "Fastsurfer - a fast and accurate deep learning based neuroimaging pipeline," *NeuroImage*, vol. 219, p. 117012, 2020.

[29] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, vol. 22, no. 268, pp. 1–8, 2021.

[30] Y.-L. Kim, K.-H. Ahn, and J.-B. Song, "Reinforcement learning based on movement primitives for contact tasks," *Robotics and Computer-Integrated Manufacturing*, vol. 62, p. 101863, 2020.

[31] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," *arXiv preprint arXiv:1606.01540*, 2016.

[32] M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, O. Pieter Abbeel, and W. Zaremba, "Hindsight experience replay," in *Advances in Neural Information Processing Systems* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), vol. 30, Curran Associates, Inc., 2017.