A DRL Approach for Mapless Planar Pushing of Arbitrary Objects in Cluttered Environments

Gabriel Santos Luz[1] **and Douglas G. Macharet**[1]

[1]Laboratório de Visão Computacional e Robótica (VeRLab)
Programa de Pós-Graduação em Ciência da Computação
Universidade Federal de Minas Gerais (UFMG)
Belo Horizonte, MG – Brazil

{gabrielluz,doug}@dcc.ufmg.br

Abstract. *Pushing is a fundamental yet challenging primitive in robotics, especially in cluttered and constrained environments. This dissertation proposes a novel two-level approach combining a low-level Deep Reinforcement Learning (DRL) policy and a high-level navigator to transport objects through narrow passages. The DRL policy ensures the object stays within a tight capsule, enabling integration with classical planners. Experiments show that this method reliably pushes irregular objects through spaces as narrow as twice their diameter, outperforming unconstrained methods, and succeeds in complex, mapless environments with dead ends and tight corridors.*

Keywords: *Planar Pushing, Mobile Robotics, Robotic Manipulation, Reinforcement Learning, Deep Learning.*

MSc. thesis defended in 06/06/2025 (PPGCC/UFMG). Committee: Prof. Douglas G. Macharet (DCC/UFMG), Prof. Luiz Chaimowicz (DCC/UFMG), Prof. Héctor lgnacio Azpúrua (DCC/UFMG), and Prof. Armando Alves Neto (DELT/UFMG).

1. Introduction

This dissertation focuses on a subfield of nonprehensile manipulation called planar pushing, which has been used throughout the literature [Bauza and Rodriguez 2017, Bauza et al. 2019, Lynch 1996, Salganicoff et al. 1993] as a testbed for developing and studying general nonprehensile methods. On a high level, planar pushing is defined as the task of positioning and orienting an object on a flat surface using only pushing actions. In addition, as it is relevant for most applications, there are obstacles on the surface that must not be touched by the robot or the object. This formulation can describe both mobile and manipulator robots. For example, a robot arm might push a cup while avoiding other kitchen utensils over a cluttered table, or a mobile robot might push a large box to transport it through a warehouse.

Furthermore, pushing is in itself a valuable manipulation primitive. It allows robots to manipulate objects that are too large, heavy, or awkwardly shaped to be grasped directly. Pushing also enables fine adjustments in object position and orientation, which can be crucial for precise assembly or preparation tasks. Dogar and Srinivasa [Dogar and Srinivasa 2010], King *et al.* [King et al. 2013], and Lee *et al.* [Lee et al. 2015] combine pushing and grasping for better manipulation, for example,

by using pushing to put an object into an easier to grasp configuration or to reduce uncertainty on the object's state. In cluttered or dynamic environments, pushing can help clear paths, reposition obstacles, or group objects for easier handling. In addition, pushing can reduce mechanical complexity and energy consumption, as it often requires simpler hardware and avoids the need for complex grasp planning or force control. It is a valuable skill for both manipulator and mobile robots that transport objects. By incorporating pushing, robots can complete more tasks, improve task efficiency, and exhibit more human-like manipulation capabilities.

Figure 1 illustrates an instance of the problem addressed. The blue circle represents the robot, the black circumference shows the boundary of the robot's local sensing range, the red polygon is the pushed object, the yellow polygon represents the goal pose, and the gray areas are the obstacles. The black dot on the object and goal helps to identify the shape's orientation. The robot must transport the object to the goal position and orientation only using pushing actions and without colliding with the obstacles. The robot does not have a map of the environment and can only perceive the obstacles inside the local sensing range.

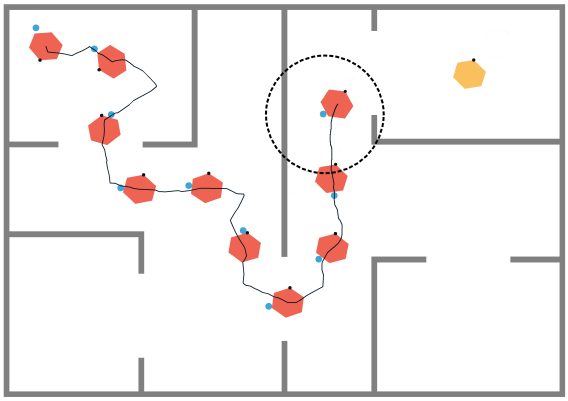


Figure 1. Mapless planar pushing: the robot (blue dot) pushes an object (red polygon) to a goal pose (yellow polygon) in an unknown and cluttered environment using only local sensing (dashed circle).

The goal is to place the object in a specific pose within the position and angle thresholds, defined by (x, y), an angle θ, and threshold distances $(\epsilon_p, \epsilon_\theta)$. Unlike the easier task of achieving a certain position without considering the orientation.

1.1. Contributions

In this dissertation, we design and implement a system that combines path planning and DRL to robustly and precisely push an object of arbitrary shape to a goal pose in a cluttered, unknown environment. To demonstrate our system's flexibility and robustness, we apply it to the challenging mapless scenario, which is useful for local obstacle avoidance and unknown environments.

Our main contributions are:

- Proposition of a DRL policy that learns to efficiently push objects of arbitrary shapes towards a predefined goal pose (x, y, θ) within a capsular region.
- A robust framework tailored for the transportation of objects in unknown cluttered environments that integrates the learned policy with a mapless local navigator.

- Design of a two-dimensional action space that allows the policy to perform both precise and efficient planar pushing movements in more constricted spaces.

  This work was also published as a peer-reviewed paper:

- Gabriel S. Luz, Douglas G. Macharet. A DRL Approach for Mapless Transportation of Arbitrary Objects. *1st Brazilian Conference on Robotics* (CROS), 2025.

2. Literature Review

Early work on planar pushing relied on analytical models based on classical mechanics [Mason 1986], but their reliance on simplified assumptions and unknown friction parameters limits real-world accuracy. To address this, researchers shifted to data-driven methods, including parameter estimation [Kloss et al. 2020], learned dynamics models [Bauza et al. 2018], and direct policy learning via Deep Reinforcement Learning (DRL) [Ferrandis et al. 2023]. DRL has proven effective in various manipulation tasks. Examples include solving a Rubik's Cube with a robotic hand [OpenAI et al. 2019], high-accuracy grasping with emergent behaviors [Kalashnikov et al. 2018], and sim-to-real dexterous manipulation [Lin et al. 2025]. In planar pushing, Zeng *et al.* [Zeng et al. 2018] showed DRL can integrate pushing and grasping in cluttered scenes. These works demonstrate DRL's potential for handling complex, diverse manipulation scenarios without manual algorithm design.

A key challenge in DRL-based pushing is obstacle avoidance. Learning both pushing and avoidance jointly is data-intensive and slow to converge. Yet, classical path planning algorithms like RRT [LaValle 1998] and PRM [Kavraki et al. 1996] are efficient and reliable. Our approach separates the tasks: a high-level planner defines a collision-free route, and a low-level DRL policy pushes the object along it, improving robustness and data efficiency. This modularity also supports extensions to dynamic or mapless planning.

Following Mandadi *et al.* [Mandadi et al. 2023] and Eoh [Eoh 2023], we decouple planning and pushing, but enhance it by constraining the DRL policy to operate within a capsule-shaped region. This allows tighter trajectory control, enabling navigation through narrower passages and closer proximity to obstacles than square-based constraints. Moreover, we tackle a harder variant of the problem: pushing to a full 3D pose and handling irregular and concave objects, outperforming prior work [Cong et al. 2022, Cho et al. 2024]. Our method is the first to push arbitrary-shaped objects to a 3D goal while avoiding obstacles and navigating through tight spaces.

Table 1 summarizes how our method compares to prior DRL approaches in terms of obstacle handling, irregular object support, 3D pose goals, real-robot validation, and generalization.

3. Problem Formulation

Given $\mathcal{E} \in \mathbb{R}^2$ an unknown static cluttered environment. Consider an object O characterized by a configuration $O_{x,y,\theta}$, where $O_{x,y} = (x, y)$ denotes its position, and O_θ represents the orientation. Let $\mathcal{R}$ be a robot represented by its position $\mathbf{p} = (x, y)$, with a kinematic model given by $\dot{\mathbf{p}} = u$.

Table 1. Comparison of DRL approaches for planar pushing across key factors. ● = fully addressed; ◐ = partially addressed; ○ = not addressed.

Paper	Obstacle avoidance	Irregular objects	3D goal pose	Real experiments	Generalizes novel objects
Clavera *et al.* [Clavera et al. 2017]	○	○	○	●	○
Peng *et al.* [Peng et al. 2018]	○	○	○	●	○
Wang *et al.* [Wang et al. 2023]	○	○	○	●	○
Ferrandis *et al.* [Ferrandis et al. 2023]	○	○	●	●	○
Ferrandis *et al.* [Ferrandis et al. 2024]	○	○	●	●	○
Cong *et al.* [Cong et al. 2022]	○	●	○	●	◐
Yang *et al.* [Yang et al. 2023]	○	●	○	●	◐
Cho *et al.* [Cho et al. 2024]	○	●	●	●	◐
Dengler *et al.* [Dengler et al. 2022]	●	○	○	○	○
Eoh [Eoh 2023]	●	○	○	○	○
Mandadi *et al.* [Mandadi et al. 2023]	●	○	●	○	○
Dengler *et al.* [Dengler et al. 2024]	●	○	●	●	○
Ours	●	●	●	○	○

Assumption 1. *The object is a rigid body of arbitrary shape, and the robot has prior knowledge of its characteristics.*

Assumption 2. *The robot has access to the poses of both the object and the goal at every time step.*

Assumption 3. *The robot is capable of distinguishing between obstacles in the environment and the object.*

Problem 1 (Planar Pushing Pose Control)**.** *A holonomic mobile robot is tasked with pushing an object O from a starting pose $S_{x,y,\theta} \in \mathbb{R}^2 \times [0, 2\pi]$ to a goal pose $G_{x,y,\theta} \in \mathbb{R}^2 \times [0, 2\pi]$ within an unknown cluttered planar environment $\mathcal{E}$. The success criteria are met if and only if the following conditions are simultaneously fulfilled before t time steps:*

$$||O_{x,y} - G_{x,y}|| < \epsilon_p \quad \text{and} \quad \Delta\theta(O_\theta, G_\theta) < \epsilon_\theta \ ,$$

where ϵ_p and ϵ_θ are error thresholds, $O_{x,y} \in \mathbb{R}^2$ is the position of the geometric center of O, and $O_\theta \in [0, 2\pi]$ is the object's orientation. The task fails if the robot or object collides with an obstacle or the time limit t is exceeded.

4. Methodology

4.1. Overview

To address the defined problem, we propose a two-fold approach. Firstly, a high-level navigator, operating without a map, dynamically constructs an obstacle-free path to the goal. The navigator selects the next subgoal based on the width of the obstacle-free capsule to it, distance to the goal, and potential information gain. Subsequently, a low-level DRL policy controls the robot to push the object toward the subgoal within the designated capsule. This iterative process continues until the object reaches the final goal. Figure 2 provides an overview of our proposed system.

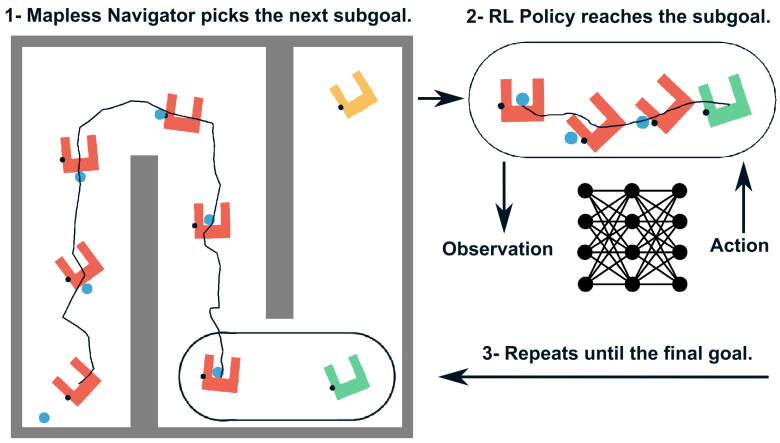


Figure 2. Overview of our mapless transportation approach. Two components alternate until the final goal is reached. The mapless navigator dynamically creates an obstacle-free path, based on past and local information, by selecting the next subgoal. Then, a Neural Network policy controls the robot to push the object to the subgoal within a capsular region.

4.2. Capsular Region

A *capsular region* is defined as the set of all points within a distance $\frac{W}{2}$ from the line segment connecting the object's start position $S_{x,y}$ to the goal position $G_{x,y}$, where $W \in \mathbb{R}$ is the capsule width. As illustrated in Figure 3, this region consists of a rectangular corridor of width W and two semicircular ends centered at $S_{x,y}$ and $G_{x,y}$. This shape provides enough room for the robot to maneuver the object in any orientation and allows complex paths to be approximated by chaining multiple capsules. Additionally, checking whether the capsule is obstacle-free is simple, as it reduces to interpolating a circle of diameter W along the path. A limitation, is that W must be large enough to contain the object in any orientation, which may be restrictive for highly elongated objects.

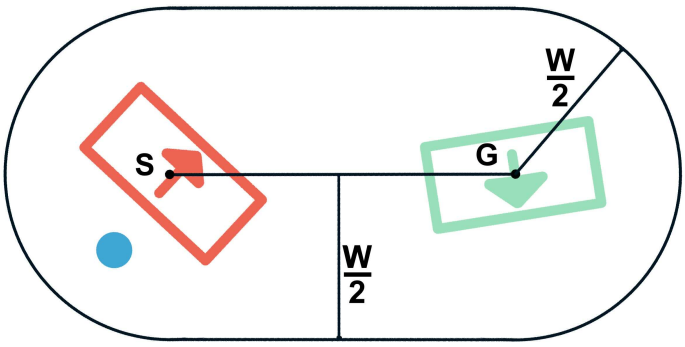


Figure 3. Example of a capsular region with a width W from the starting object position S to the goal G. The robot is depicted as a blue circle.

4.3. Planar Pushing within a Capsular Region with DRL

To transport an object through a sequence of capsules, the pushing policy must ensure that both the robot and object remain within the capsular region until reaching each subgoal. Simply learning to push toward a goal is insufficient, as efficient paths may

leave the region. To address this, we designed a DRL training environment with a custom reward function that enforces capsule compliance. In each episode, the object and goal are randomly placed within a rectangular space, and the agent starts near the object.

4.3.1. Action Space

We model the robot as a holonomic circular agent. The policy outputs a continuous action $a = (a_\rho, a_\theta) \in [0, 1]^2$, where a_θ sets the movement direction and a_ρ determines the step length between predefined bounds $[min_L, max_L]$. The resulting vector v defines a target point in the robot's local frame, and the robot moves toward this point using constant velocity until it's reached or a step limit is hit.

4.3.2. Observation Space

Following Ferrandis *et al.* [Ferrandis et al. 2023], the observation includes only the object's pose, not its shape, making the policy shape-specific but simplifying the problem focus. To support generalization across varying capsule widths W, which are randomly sampled, the observation includes W as part of a goal-conditioned setup [Liu et al. 2022]. The observation, expressed in the agent's frame, contains 9 normalized values: the polar coordinates and orientation of the object, its start position, the goal pose, and the capsule width. This compact representation allows the agent to adapt to both the object's configuration and the varying task constraints.

4.3.3. Reward Function

An episode ends in success when the object reaches the goal pose within set tolerances, or in failure if the object or robot exits the capsule or the robot strays too far from the object. A small time penalty encourages efficient trajectories. To guide learning, we apply potential-based reward shaping [Ng et al. 1999], using two potential functions: one for reducing the distance to the goal position and another for aligning orientations. The shaped reward is scaled so its total contribution averages 1.0 per successful episode, with termination rewards of +1.0 for success and -1.0 for failure. The final reward is the sum of the termination reward, the potential reward, and a constant time penalty of -0.01.

4.4. Mapless Transportation

The proposed DRL policy can be paired with any planner that builds a path from connected obstacle-free capsules. To demonstrate its flexibility, we apply it in a mapless navigation scenario using only local sensing, where the robot builds the path dynamically and may need to backtrack. Our local planner samples candidate subgoals from a ring around the object, ranks them using a heuristic score combining viability, information gain, and proximity to the goal, and selects the best one. The robot uses 360° sensing (*e.g.*, laser) to detect obstacles. Once a subgoal is set, the DRL policy attempts to reach it within 30 steps. Orientation is adjusted in 30° increments toward the goal. To handle tight spaces, if the object can't reach the subgoal precisely, we initialize the next capsule from the closest point on the previous capsule's line segment to ensure continuity.

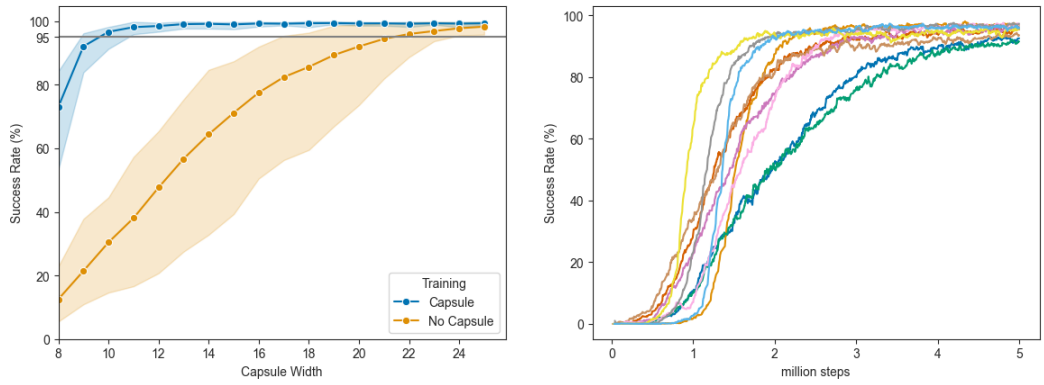
5. Experiments and Results

We performed simulated experiments to evaluate the pushing policy across different capsule widths and assess the full mapless transportation system in cluttered environments of varying complexity. Ten distinct policies were trained and tested, each specialized in pushing a different object with an arbitrary shape. The results demonstrate that our approach can robustly handle complex geometries and integrate seamlessly with a mapless navigation strategy. Video demonstrations are available in the footnote[1].

5.1. Planar Pushing within a Capsular Region with DRL

We evaluated the minimum capsule width W required for reliable pushing by training and testing ten DRL policies, each specialized in pushing a different arbitrarily shaped object. To enable generalization across widths, we trained each policy with $W \sim [8, 15]$ and evaluated success rates over varying widths. Training used the SAC algorithm for 5M steps, with reward scaling and periodic evaluation.

Results (Figure 4(a), Table 2) show that capsule-constrained policies achieve over 95% success for $W \geq 10\,\text{m}$, allowing transport through passages twice the object's diameter. In contrast, unconstrained policies require $W \geq 22\,\text{m}$, *i.e.*, 4.4 times the object's size. Training curves (Figure 4(b)) confirm convergence for all objects.



(a) (b)

Figure 4. (a) Success rate (%) for different capsule widths (m) over 1,000 episodes for each policy. (b) Smoothed success rates plotted over training steps for each of the 10 policies.

Table 2. Success rate (%) for different capsule widths (m) over 1,000 episodes for policies trained with and without the capsule restriction. We take the average result and the 95% confidence interval over the 10 policies.

	10m	15m	20m	25m
Capsule	96.7 ± 1.5	99.0 ± 0.5	99.3 ± 0.3	99.4 ± 0.2
No Capsule	30.4 ± 6.8	71.2 ± 9.6	92.1 ± 5.2	98.3 ± 0.9

[1] Videos of the executions: https://youtu.be/8sixF2if-tc

5.2. Mapless Transportation

We evaluated the mapless transportation system on five obstacle maps (Figure 5) with varying complexity. Each policy was tested across 200 trials per map version, with random start and goal poses. The local navigator uses 360° laser sensing and samples 200 subgoal candidates per step.

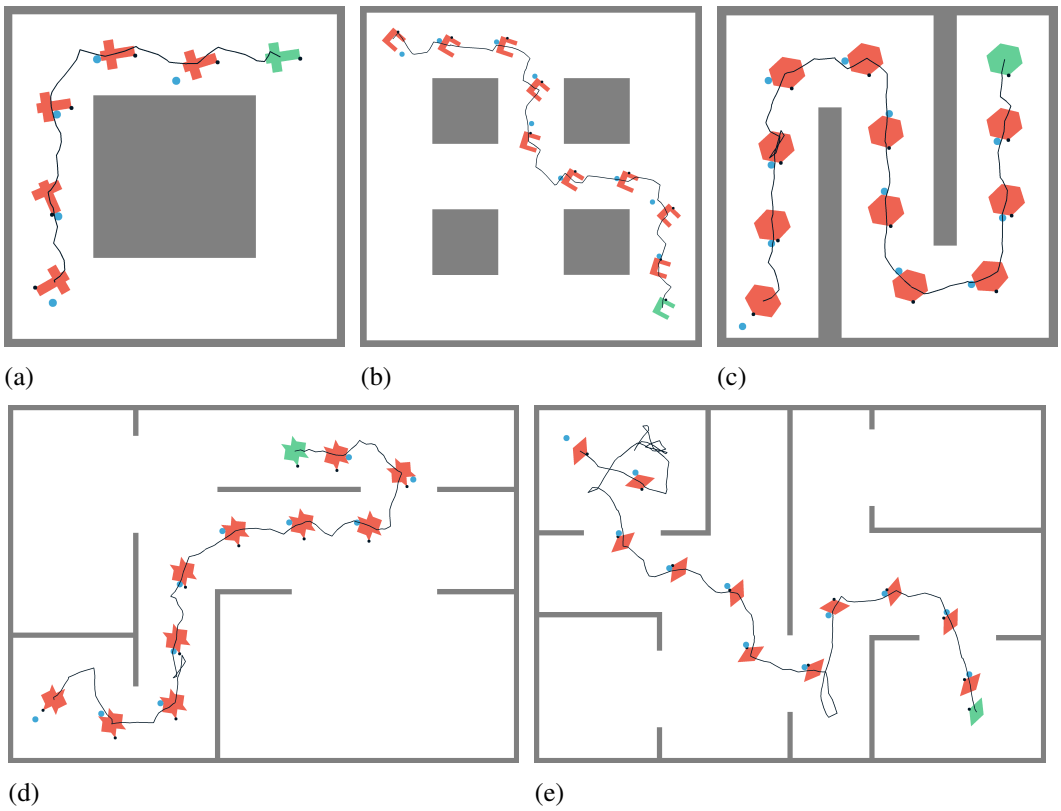


Figure 5. Overall scenarios with 25m passages.

Table 3 shows that the system performs reliably in environments with passages $\geq$25m wide, achieving over 96% success. Performance drops slightly in 20m passages due to navigation challenges, though the low-level policy can still handle them.

Table 3. Success rate over 200 trials for each combination of map version and object. We take the average result and the 95% confidence interval over the 10 objects.

	Map 1	Map 2	Map 3	Map 4	Map 5
20m	86.9 ± 5.9	87.6 ± 5.0	82.5 ± 6.5	91.6 ± 3.9	79.8 ± 7.0
25m	98.1 ± 1.4	97.2 ± 2.2	96.6 ± 2.9	97.2 ± 1.4	96.0 ± 2.7
30m	99.1 ± 0.6	98.7 ± 1.1	98.4 ± 1.3	98.2 ± 1.1	97.9 ± 1.3

Figure 6 illustrates a very challenging case where the robot explored most of the map, navigating dead-ends and tight turns before succeeding.

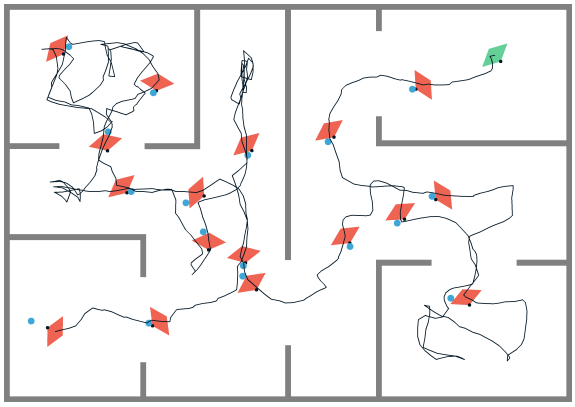


Figure 6. A challenging scenario where the robot needed to explore the entire map before reaching the final goal.

6. Conclusions and Future Work

6.1. Conclusions

Deep Reinforcement Learning (DRL) has shown strong results in object manipulation [OpenAI et al. 2019, Kalashnikov et al. 2018, Lin et al. 2025], and prior work [Ferrandis et al. 2023, Zeng et al. 2018, Cho et al. 2024] has demonstrated its potential for planar pushing. However, most DRL approaches overlook pushing among obstacles, crucial for real-world tasks—and focus only on simple shapes or 2D goal positions [Mandadi et al. 2023, Eoh 2023, Dengler et al. 2022].

This work proposes a two-level approach that combines a DRL policy and a high-level navigator to push arbitrary objects in cluttered, unknown environments. The DRL policy pushes toward subgoal poses while keeping the object within a capsule-shaped region, enabling safe and predictable motion. This allows seamless integration with classical planners to navigate through narrow passages. Simulated experiments with diverse objects show that our capsule-based policy reliably pushes within regions just twice the object's diameter, approaching the theoretical $1.4\times$ lower bound and outperforming the $4.4\times$ baseline. When combined with a mapless navigator, the system consistently transports objects in complex environments with dead-ends and tight corridors.

6.2. Future Work

Future work should address limitations that hinder real-world applicability of the proposed method. The current policy lacks adaptability to variations in object and surface properties, which significantly affect pushing dynamics [Yu et al. 2016, Bauza et al. 2019]. A more general solution would allow zero-shot manipulation of diverse objects and rapid adaptation to novel scenarios, potentially achievable through visual/tactile sensing, few-shot learning, and meta-learning. The capsule constraint also presents limitations, being overconservative and unsuitable for elongated objects. Allowing the policy to perceive local obstacles could enable more efficient pushing while maintaining the benefits of task decomposition [Dengler et al. 2022]. Additionally, recent advances in general manipulation using Diffusion Policies [Chi et al. 2023] and Vision-Language-Action models [Pertsch et al. 2025] offer promising paths forward. These

models show strong generalization and can handle complex objects and tasks, and could be combined with DRL for enhanced adaptability and precision.

References

Bauza, M., Alet, F., Lin, Y.-C., Lozano-Pérez, T., Kaelbling, L. P., Isola, P., and Rodriguez, A. (2019). Omnipush: accurate, diverse, real-world dataset of pushing dynamics with rgb-d video. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4265–4272. IEEE.

Bauza, M., Hogan, F. R., and Rodriguez, A. (2018). A data-efficient approach to precise and controlled pushing. In *Conference on Robot Learning*, pages 336–345. PMLR.

Bauza, M. and Rodriguez, A. (2017). A probabilistic data-driven model for planar pushing. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3008–3015. IEEE.

Chi, C., Xu, Z., Feng, S., Cousineau, E., Du, Y., Burchfiel, B., Tedrake, R., and Song, S. (2023). Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, page 02783649241273668.

Cho, Y., Han, J., Cho, Y., and Kim, B. (2024). Corn: Contact-based object representation for nonprehensile manipulation of general unseen objects. *arXiv preprint arXiv:2403.10760*.

Clavera, I., Held, D., and Abbeel, P. (2017). Policy transfer via modularity and reward guiding. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1537–1544. IEEE.

Cong, L., Liang, H., Ruppel, P., Shi, Y., Görner, M., Hendrich, N., and Zhang, J. (2022). Reinforcement learning with vision-proprioception model for robot planar pushing. *Frontiers in Neurorobotics*, 16:829437.

Dengler, N., Ferrandis, J. D. A., Moura, J., Vijayakumar, S., and Bennewitz, M. (2024). Learning goal-directed object pushing in cluttered scenes with location-based attention. *arXiv preprint arXiv:2403.17667*.

Dengler, N., Großklaus, D., and Bennewitz, M. (2022). Learning goal-oriented non-prehensile pushing in cluttered scenes. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1116–1122. IEEE.

Dogar, M. R. and Srinivasa, S. S. (2010). Push-grasping with dexterous hands: Mechanics and a method. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2123–2130. IEEE.

Eoh, G. (2023). Deep-reinforcement-learning-based object transportation using task space decomposition. *Sensors*, 23(10):4807.

Ferrandis, J. D. A., Moura, J., and Vijayakumar, S. (2023). Nonprehensile planar manipulation through reinforcement learning with multimodal categorical exploration. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5606–5613. IEEE.

Ferrandis, J. D. A., Moura, J., and Vijayakumar, S. (2024). Learning visuotactile estimation and control for non-prehensile manipulation under occlusions. *arXiv preprint arXiv:2412.13157*.

Kalashnikov, D., Irpan, A., Pastor, P., Ibarz, J., Herzog, A., Jang, E., Quillen, D., Holly, E., Kalakrishnan, M., Vanhoucke, V., and Levine, S. (2018). Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation. *ArXiv*, abs/1806.10293.

Kavraki, L. E., Svestka, P., Latombe, J.-C., and Overmars, M. H. (1996). Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE transactions on Robotics and Automation*, 12(4):566–580.

King, J. E., Klingensmith, M., Dellin, C. M., Dogar, M. R., Velagapudi, P., Pollard, N. S., and Srinivasa, S. S. (2013). Pregrasp manipulation as trajectory optimization. In *Robotics: Science and Systems*. Berlin.

Kloss, A., Bauza, M., Wu, J., Tenenbaum, J. B., Rodriguez, A., and Bohg, J. (2020). Accurate vision-based manipulation through contact reasoning. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6738–6744. IEEE.

LaValle, S. (1998). Rapidly-exploring random trees: A new tool for path planning. *Research Report 9811*.

Lee, G., Lozano-Pérez, T., and Kaelbling, L. P. (2015). Hierarchical planning for multi-contact non-prehensile manipulation. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 264–271. IEEE.

Lin, T., Sachdev, K., Fan, L., Malik, J., and Zhu, Y. (2025). Sim-to-real reinforcement learning for vision-based dexterous manipulation on humanoids. *arXiv preprint arXiv:2502.20396*.

Liu, M., Zhu, M., and Zhang, W. (2022). Goal-conditioned reinforcement learning: Problems and solutions. *arXiv preprint arXiv:2201.08299*.

Lynch, K. M. (1996). *Nonprehensile robotic manipulation: Controllability and planning*. Carnegie Mellon University.

Mandadi, V. R., Saha, K., Guhathakurta, D., Qureshi, M. N., Agarwal, A., Sen, B., Das, D., Bhowmick, B., Singh, A., and Krishna, M. (2023). Disentangling planning and control for non-prehensile tabletop manipulation. In *2023 IEEE 19th International Conference on Automation Science and Engineering (CASE)*, pages 1–6. IEEE.

Mason, M. T. (1986). Mechanics and planning of manipulator pushing operations. *The International Journal of Robotics Research*, 5(3):53–71.

Ng, A. Y., Harada, D., and Russell, S. (1999). Policy invariance under reward transformations: Theory and application to reward shaping. In *Icml*, volume 99, pages 278–287. Citeseer.

OpenAI, Akkaya, I., Andrychowicz, M., Chociej, M., teusz Litwin, M., McGrew, B., Petron, A., Paino, A., Plappert, M., Powell, G., Ribas, R., Schneider, J., Tezak, N. A., Tworek, J., Welinder, P., Weng, L., Yuan, Q., Zaremba, W., and Zhang, L. M. (2019). Solving rubik's cube with a robot hand. *ArXiv*, abs/1910.07113.

Peng, X. B., Andrychowicz, M., Zaremba, W., and Abbeel, P. (2018). Sim-to-real transfer of robotic control with dynamics randomization. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 3803–3810. IEEE.

Pertsch, K., Stachowicz, K., Ichter, B., Driess, D., Nair, S., Vuong, Q., Mees, O., Finn, C., and Levine, S. (2025). Fast: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*.

Salganicoff, M., Metta, G., Oddera, A., and Sandini, G. (1993). *A vision-based learning method for pushing manipulation*, volume 54. University of Pennsylvania.

Wang, S., Sun, L., Zha, F., Guo, W., and Wang, P. (2023). Learning adaptive reaching and pushing skills using contact information. *Frontiers in Neurorobotics*, 17:1271607.

Yang, M., Lin, Y., Church, A., Lloyd, J., Zhang, D., Barton, D. A., and Lepora, N. F. (2023). Sim-to-real model-based and model-free deep reinforcement learning for tactile pushing. *IEEE Robotics and Automation Letters*, 8(9):5480–5487.

Yu, K.-T., Bauza, M., Fazeli, N., and Rodriguez, A. (2016). More than a million ways to be pushed. a high-fidelity experimental dataset of planar pushing. In *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 30–37. IEEE.

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