MAC: Morphology-Aware Control Design with Deep Reinforcement Learning for Navigation of Squeezable Unmanned Aerial Vehicles

Van Huyen Dang, Adrian Redder, Erdal Kayacan
Automatic Control Group (RAT)
Department of Electrical Engineering and Information Technology
Paderborn University, Paderborn, Germany
{vhdang, aredder, erdal.kayacan}@upb.de

Abstract: Traditional methods optimize robot morphology independently using evolutionary algorithms, where fitness functions are applied to evaluate each genotype. Control strategies are then designed based on this predefined morphology. While the concept of co-design, considering multiple aspects simultaneously, is not new, it is often impractical due to the time-consuming process of manufacturing new morphology. A common solution to this limitation is restricting the design space to morphology with similar topologies. In this extended abstract, we investigate a squeezable unmanned aerial vehicle (UAV) as the controlled plant and propose a novel morphology-aware control (MAC) method leveraging deep reinforcement learning (DRL) to solve a vision-based navigation problem under extreme settings in this study. Our approach integrates the agent's morphology directly into the learning process of the control policy, enabling fast morphology-policy co-design. To simplify the control problem, we restrict the quadrotor configuration that can be transformed only between X and H via a single squeezing angle (ξ). The quadrotor's body shape can be horizontally reduced to 52.4 $\%$ of its original shape. Simulation results show that the navigation policy trained with an understanding of morphology is more effective.

Keywords: Unmanned Aerial Vehicle (UAV), Deep reinforcement learning, Morphology-policy co-design

1 Introduction

UAVs are used for a wide range of applications, e.g., inspection, exploration [1, 2, 3], and search and rescue [4, 5, 6]. It is commonly seen that the controllers are often designed with fixed morphology that is assumed to be pre-defined and unchangeable for a specific task. However, morphology and controller are strongly coupled, and any morphological change, due to changes in environments, will influence existing controllers. This results in a need for a co-design method that considers multiple aspects simultaneously to design re-configurable or adaptable robots with learning capabilities. Early works on co-design problems often leverage evolutionary robotics to optimize both morphology and controller [7, 8, 9]. Alternatively, recent research focuses on the development of artificial intelligence-empowered robots that can perceive their environment and make decisions based on what they perceive. Towards this end, reinforcement learning has gained significant attention in the field of robotics because it enables robots to autonomously learn and refine their behavior according to changes in the environment, thereby enhancing their overall adaptability and robustness [10, 11, 12]. However, these works are particularly applied to legged locomotion. For operating in extreme environments where ground robots might not work effectively, more agile robots, such as squeezable UAVs, [13, 14], have emerged as a promising solution. Building on that need and the latest trends in robotics, we propose a morphology-policy co-design method for squeezable UAVs

Workshop on Morphology-Aware Policy and Design Learning (MAPoDeL) at CoRL 2024.

using DRL. In particular, we emphasize a novel end-to-end morphology-aware navigation policy that operates solely on visual sensory data.

The remaining parts of this manuscript are organized as follows. Section 2 describes relevant works about morphing mechanisms and control strategies regarding to these mechanisms. Section 3 presents the proposed MAC method for navigating a squeezable drone. Section 4 describes briefly the simulation setup and results. Finally, Section 5 provides the conclusions and future works.

2 Related works

In this section, we present relevant works dealing with the morphology-policy co-design problem of aerial robots. We first investigate morphing mechanisms as described in Sec. 2.1, and control strategies, commonly utilized for controlling the agent due to these changes, are further discussed in Sec. 2.2.

2.1 Morphing mechanism of aerial robots

Previous studies have explored the development of morphing mechanisms that enable aerial robots to adapt their morphology for various functional advantages. These designs enhance gliding efficiency across a broad velocity range [15] and allow for flight through tighter spaces [16, 13, 14]. For instance, [16] presented an X-morf drone that contains two independent arms. These two arms are actuated by a single scissor joint that allows to reduce a standard X configuration up to 28.5 %. [13] presented a large drone equipped with a folding mechanism capable of navigating cluttered environments by reducing its wingspan by 48%, allowing it to pass through narrow vertical gaps at 2.5 m/s. To further enhance versatility, [14] introduced a sophisticated morphing design for quadrotors, featuring four independently rotating arms. This design allows the quadrotor to shift from a conventional X configuration to alternative configurations—such as H, O, or T—tailored for specific tasks. Alternatively, [17] developed a morphing aerial robot with a two-dimensional multilink structure that not only enables stable transformations but also allows the robot to function as a complete gripper. In this study, we choose a squeezing drone constructed as presented in [14] and focus on developing a morphology-aware policy using the DRL framework. To simplify the control problem, we restrict the quadrotor configuration that can be transformed only between X and H configuration via a single squeezing angle ξ. The squeezable drone is designed as shown in Fig. 1. The designed drone in this work can morph up to 52.4 % of its body shape horizontally, so this mechanism allows the drone can pass through vertical gaps that are smaller than the size of the drone.

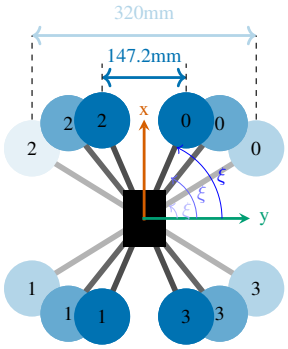


Figure 1: Construction of the squeezable drone. The four rotating arms are identical, allowing them to be controlled simultaneously by a single variable. The body shape can be continuously reduced to 52.4 % of its original shape.

2.2 Control strategies

To control a morphing drone, common strategies divide the system into three key modules: perception, planning, and control. Most approaches focus primarily on control design, assuming that the perception and planning modules are well-designed. The control module often relies on model-based techniques. For instance, Model Reference Adaptive Control (MRAC) has been employed to manage uncertainties in inertia and the center of mass due to reconfiguration during flight, as shown in [16]. In another example, [13] introduced a squeezing angle as an additional degree of freedom and implemented position and attitude control using two PID controllers. To handle four extra degrees of freedom, [14] used a Linear Quadratic Regulator (LQR) controller, which recalculates the inertial matrix and center of gravity according to morphological adjustments.

Moving beyond model-based methods, model-free approaches present promising alternatives. Inspired by biological systems, where animals adapt their body shapes in response to environmental changes based primarily on visual cues, these methods leverage DRL to incorporate morphology into policy learning. However, most recent works have applied these methods in legged locomotion [10, 11, 12]. Thus, we develop a MAC strategy using DRL to train a navigation policy for the drone. This approach solves navigation in an end-to-end manner, factoring in morphological changes during policy training. In essence, the drone perceives environmental changes through a camera and adjusts its body shape to navigate optimally.

3 Methodology

We propose an end-to-end solution for vision-based navigation that integrates the drone's morphology into policy learning. The system architecture, shown in Fig. 2, enables the DRL agent to receive images as input and make navigation decisions based on visual perception. Notably, the agent is aware of its morphing ability, allowing it to adjust its shape to pass through narrow gaps. We train this DRL agent using the deep deterministic policy gradient (DDPG) algorithm [18], which is well-suited for continuous action spaces. Key considerations include: (1) how to formulate the co-design problem to incorporate morphology, and (2) the optimal policy architecture.

3.1 Morphology-aware co-design with reinforcement learning

We consider the navigation problem as an Markov decision process (MDP) with state space represented by the local observation space, while the global environment dynamics are viewed as part of the MDP uncertainty. Hence, the problem can be formulated as a MDP described by a tuple (S, A, p, r, γ) in which S, A, p, r, γ denote state, action, transition model, reward, and discount factor, respectively. Therefore, a maximization problem can be written as (1).

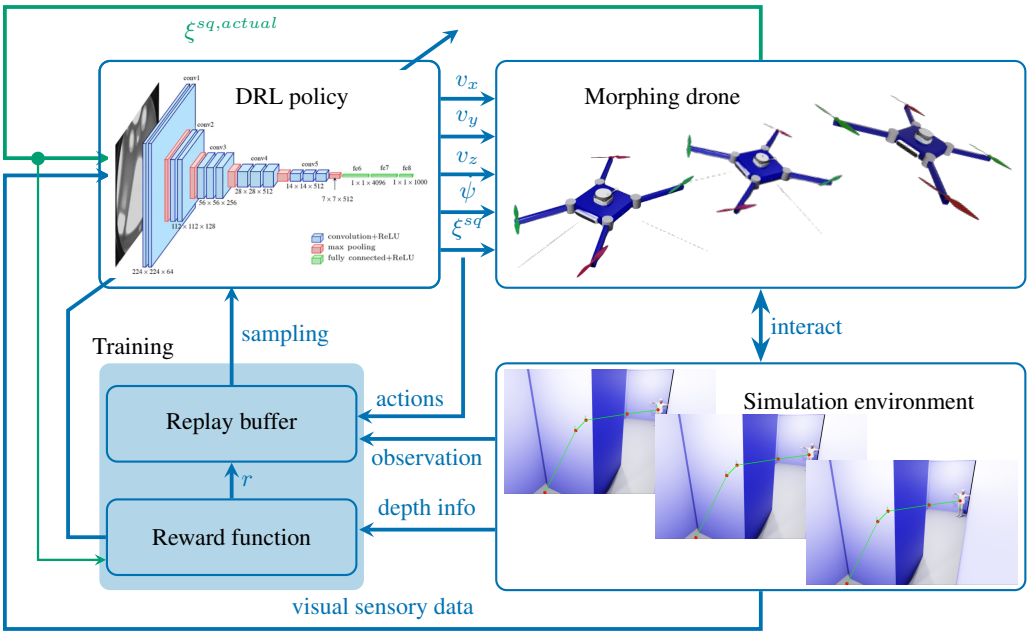


Figure 2: The proposed MAC method incorporates the agents' morphology in training the navigation policy. To meet this need for realistic training environments for a model learning algorithm, this work develops an DRL vision-based navigation benchmark by integrating high-fidelity simulation in the NVIDIA Isaac Sim with ROS. This benchmark can be used efficiently and with minimal effort to train and test DRL vision-based navigation algorithm.

$$\max_{\theta^{\pi}} \mathbb{E}\left(Q^{*}_{\theta^{\phi}}(s, \pi_{\theta^{\pi}}(s, \xi^{sq}))\right) \tag{1}$$

In the DRL framework, the performance of DRL algorithms heavily depends on its design. Typically, reward functions include morphology-dependent components, but this approach becomes impractical when exploring a large morphological design space. In this work, we consider squeezable UAVs to be a morphing agent whose design space can be restricted to squeezing angle (ξ_{sq}) only. Thus, we train the agent policy $\pi_{\theta^{\pi}}(s, \xi^{sq})$ that outputs actions $a \in \pi_{\theta^{\pi}}(s, \xi^{sq})$ to maximize the state action value $Q_{\theta^{\phi}}(s, a)$. Notably, the agents' morphology information is involved in the learning process. The policy and state-action functions are represented by deep neural networks characterized by the parameter sets θ^{π} and θ^{ϕ}, respectively. The formulation described by (1) implies that if we know the optimal state-action function, we can determine the optimal action $\pi^{*}_{\theta}(s)$ in any given state s. Therefore, it is important to train the critic network that ensures $Q_{\theta^{\phi}}(s, a)$ closer to the optimal action-value function $Q^{*}_{\theta^{\phi}}(s, a)$. This can be done by minimizing the mean-squared error described by

$$L(\theta^{\phi}, \mathcal{R}) = \mathbb{E}_{(s_t, \xi_{sq}, a_t, r_t, s_{t+1}, T) \sim \mathcal{R}} \left[\left(Q_{\theta^{\phi}}(s_t, a_t) - Q_{\text{target}} \right)^2 \right] \tag{2}$$

where

$$Q_{\text{target}} = \left[r_t + \gamma(1-d) Q_{\theta^{\phi}_{\text{target}}}(s_{t+1}, \pi_{\theta^{\pi}_{\text{target}}}(s_{t+1}) \right] \tag{3}$$

and $\mathcal{R}$ denotes an experience replay buffer. The variable $d \in \{0, 1\}$ indicates whether the state terminated or not. In DDPG [18], target critic $Q_{\theta^{\phi}_{\text{target}}}$ and policy $\pi_{\theta^{\pi}_{\text{target}}}$ networks are used to stabilize the training process by introducing a delay when updating target networks.

3.2 Policy network architecture

The network architecture of the policy is depicted in Fig. 3. We define actor and critic networks, whose architecture includes variational autoencoder (VAE) followed by two fully connected layers. This architecture takes raw depth images and actual squeezing angles and outputs directly linear velocities, yaw rate, and squeezing actions.

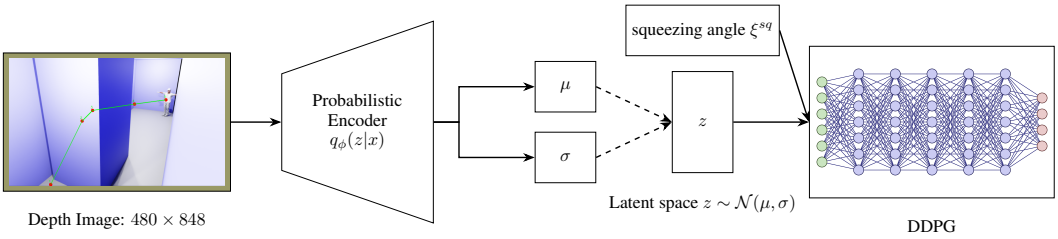


Figure 3: The architecture of the policy network employs a VAE [19] to encode images into a latent space. We utilize the VAE model trained by [20]. Subsequently, we freeze the weights of the VAE model and only train the actor and critic networks of the DDPG.

The encoder receives a depth image input of size 480×848 and outputs a latent space vector with a dimension of 64×1. This latent space will be concatenated with the squeezing angle to form the input to the DDPG. The output of the DDPG is a vector of 5 actions, including linear velocities, yaw rate, and squeezing actions. The common hyperparameters, used for training the actor and critic networks of these methods, are given in Table 1. The fully connected layers 1 and 2 for both actors and critics are defined as 400, and 300, respectively.

4 Simulation study

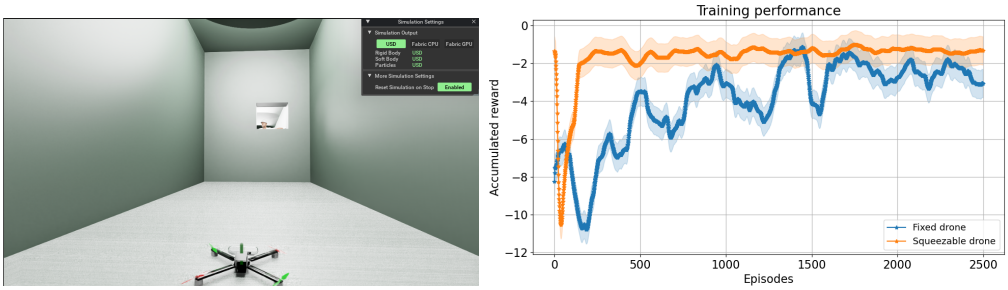
4.1 Scenario setup

To validate our proposed approach, we design a scenario where the drone is requested to navigate a hallway to search for a person in the presence of walls and obstacles that define a narrow gap,

shown in Fig. 4a. The gap is defined with a dimension of the width and height $500mm \times 500mm$. The width is slightly larger than the full wingspan of the agent, as described in Fig. 1. To prove the concept of the co-design problem, the agent is expected to adjust its body in order to improve the success rate of passing through that narrow gap.

4.2 Training results

The training is conducted over 2500 episodes, each running for 1000 iterations. We use fixed and squeezable drones to evaluate our proposed MAC method. The training environment is randomized at the start of each episode. Figure 4b shows that the agent with the squeezable mechanism achieves higher rewards compared to the fixed one. It implies that the agent with the squeezable mechanism navigates through narrow gaps more easily and reaches the target person more effectively.



(a) Simple setting

(b) Training results

Figure 4: We train both the fixed and squeezable drones for 2,500 episodes, running the algorithms five times using different random seeds.

Table 1: Training parameters

Parameters	**Values**	**Description**
η_a	0.25e-3	Actor learning rate
η_c	0.25e-2	Critic learning rate
ρ	0.99	Polyak factor
γ	0.95	Discounted factor
ω_c	1e-4	Weight decay factor for the critic
b	64	Batch size
M_{rb}	500000	Memory size of the replay buffer

5 Conclusion

In this extended abstract, we present a proof of concept for a morphology-aware control method utilizing the DRL framework applied to control a squeezable drone. We incorporate the agent's morphology during the training of the navigation policy. This co-design approach enhances the trained policy, enabling the squeezable drone to complete tasks more effectively. In the future, we plan to do more experiments and compare our method with other approaches that also incorporate the squeezing angle into their control policies. Additionally, we will increase the training environment's complexity to analyze data efficiency and learning convergence issues. Subsequently, we will evaluate the proposed MAC algorithm in real time through sim-to-real transfer experiments.

References

[1] T. Dang, F. Mascarich, S. Khattak, C. Papachristos, and K. Alexis. Graph-based path planning for autonomous robotic exploration in subterranean environments. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3105–3112. IEEE, 2019.

[2] K. Alexis. Resilient autonomous exploration and mapping of underground mines using aerial robots. In *2019 19th International Conference on Advanced Robotics (ICAR)*, pages 1–8. IEEE, 2019.

[3] M. Kulkarni, M. Dharmadhikari, M. Tranzatto, S. Zimmermann, V. Reijgwart, P. De Petris, H. Nguyen, N. Khedekar, C. Papachristos, L. Ott, et al. Autonomous teamed exploration of subterranean environments using legged and aerial robots. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 3306–3313. IEEE, 2022.

[4] M. B. Bejiga, A. Zeggada, A. Nouffidj, and F. Melgani. A convolutional neural network approach for assisting avalanche search and rescue operations with uav imagery. *Remote Sensing*, 9(2):100, 2017.

[5] K. G. Panda, S. Das, D. Sen, and W. Arif. Design and deployment of uav-aided post-disaster emergency network. *IEEE Access*, 7:102985–102999, 2019.

[6] N. Zhao, W. Lu, M. Sheng, Y. Chen, J. Tang, F. R. Yu, and K.-K. Wong. Uav-assisted emergency networks in disasters. *IEEE Wireless Communications*, 26(1):45–51, 2019.

[7] J. E. Auerbach and J. C. Bongard. Environmental influence on the evolution of morphological complexity in machines. *PLoS computational biology*, 10(1):e1003399, 2014.

[8] K. Miras, E. Ferrante, and A. E. Eiben. Environmental influences on evolvable robots. *PloS one*, 15(5):e0233848, 2020.

[9] T. F. Nygaard, C. P. Martin, E. Samuelsen, J. Torresen, and K. Glette. Real-world evolution adapts robot morphology and control to hardware limitations. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 125–132, 2018.

[10] D. Ha. Reinforcement learning for improving agent design. *Artificial life*, 25(4):352–365, 2019.

[11] C. Schaff, D. Yunis, A. Chakrabarti, and M. R. Walter. Jointly learning to construct and control agents using deep reinforcement learning. In *2019 international conference on robotics and automation (ICRA)*, pages 9798–9805. IEEE, 2019.

[12] K. S. Luck, H. B. Amor, and R. Calandra. Data-efficient co-adaptation of morphology and behaviour with deep reinforcement learning. In *Conference on Robot Learning*, pages 854–869. PMLR, 2020.

[13] V. Riviere, A. Manecy, and S. Viollet. Agile robotic fliers: A morphing-based approach. *Soft robotics*, 5(5):541–553, 2018.

[14] D. Falanga, K. Kleber, S. Mintchev, D. Floreano, and D. Scaramuzza. The foldable drone: A morphing quadrotor that can squeeze and fly. *IEEE Robotics and Automation Letters*, 4(2): 209–216, 2018.

[15] D. Lentink, U. K. Müller, E. Stamhuis, R. De Kat, W. Van Gestel, L. Veldhuis, P. Henningsson, A. Hedenström, J. J. Videler, and J. L. Van Leeuwen. How swifts control their glide performance with morphing wings. *Nature*, 446(7139):1082–1085, 2007.

[16] A. Desbiez, F. Expert, M. Boyron, J. Diperi, S. Viollet, and F. Ruffier. X-morf: A crash-separable quadrotor that morfs its x-geometry in flight. In *2017 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED-UAS)*, pages 222–227. IEEE, 2017.

[17] M. Zhao, K. Kawasaki, X. Chen, S. Noda, K. Okada, and M. Inaba. Whole-body aerial manipulation by transformable multirotor with two-dimensional multilinks. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5175–5182. IEEE, 2017.

[18] 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.

[19] D. Hoeller, L. Wellhausen, F. Farshidian, and M. Hutter. Learning a state representation and navigation in cluttered and dynamic environments. *IEEE Robotics and Automation Letters*, 6 (3):5081–5088, 2021.

[20] M. Kulkarni and K. Alexis. Reinforcement learning for collision-free flight exploiting deep collision encoding. *arXiv preprint arXiv:2402.03947*, 2024.