Reinforcement Learning for Agile Flight: From Perception to Action

Yunlong Song, Davide Scaramuzza
University of Zurich

***Abstract*—Modern aerial robotics are increasingly being called upon to execute complex, agile maneuvers in dynamically changing environments. At the crux of this challenge lies the integration of robust perception systems with adaptive control mechanisms. Leading-edge solutions, exemplified by the Skydio drone, predominantly rely on model-based methodologies. In contrast, this work explores a different paradigm: Reinforcement Learning (RL). Beginning with the introduction of a flexible simulator and a versatile control framework, we highlight the role of simulation for learning and the need for a versatile platform for development. Next, we show how to leverage the advantage of both policy search and model predictive control to develop a learning-based controller for agile flight in a dynamic environment. Furthermore, we delve deep into a fundamental study between optimal control and RL and show that the fundamental advantage of RL over optimal control lies in its capability to optimize task-level objectives directly. Our findings allow us to achieve an important milestone in robotics: pushing a super agile drone to its physical limit, achieving a maximum acceleration of 12g. Finally, we close the loop by showing how to leverage visual representation learning for vision-based control.**

A list of video demonstrations:

- Reaching the limit in autonomous drone racing
- Flying through dynamic gates
- Navigating through cluttered environment
- Perception-aware flight
- Robust representation learning
- Flightmare simulation

I. Introduction

Autonomous navigation of micro aerial vehicles has recently achieved impressive results outside of research labs, from exploring Mars to swarm navigation [1]. For maximal performance, some tasks require flying the vehicle at high speeds and pushing the aircraft to its physical limits of speed and acceleration. In those scenarios, tolerance for error is low: a small mistake can lead to a catastrophic crash. A fundamental question is how should we design a control system for flying fast while being robust against unknown disturbances.

Reinforcement learning [2] is an attractive approach and has demonstrated exceptional performance in various robotic domains, such as quadrupedal locomotion over challenging terrain [3], [4]. RL has several advantages over model-based control. First, it learns a control policy via offline optimization, enabling the trained policy to compute control commands during deployment efficiently. Second, RL can directly optimize the task's performance objective, eliminating the need for explicit intermediate representations such as trajectories. Finally, RL is modeled as a Markov Decision Process, where the state transition model can be formulated via probability, enabling learning of a stochastic policy that is effective in diverse environments.

Applying reinforcement learning (RL) to agile flight is challenging due to the need for vast training data, the reality gap between simulation and the real world, and safety risks during high-speed flight. In addition, hardware limitations and communication latencies further complicate the real-time application of RL in agile flight environments.

II. Contributions

In this study, we explore the intersection of high-speed simulation, model predictive control, reinforcement learning, and representation learning, aimed at addressing the challenges in vision-based flight. Our results are summarized in Fig. 1.

First, we introduced two versatile software tools: Flightmare and Agilicious. Flightmare [5] is a tailored simulation tool for quadrotor applications, focusing on high-speed simulation through parallel processing and photorealistic rendering. Agilicious [6] is a co-designed hardware and software framework for testing model-based and neural-network-based systems in the real world with a physical system.

Second, we proposed novel algorithms to design effective planning and control algorithms for agile flight. This includes a policy-search-for-model-predictive-control framework [7], [8], an end-to-end control method [9], [10], and a hybrid method [11] that combines classical topological path planning with model-free deep reinforcement learning. We demonstrated the effectiveness of our approaches with several challenging tasks, including flying through dynamic gates, pushing the limit in autonomous drone racing, and navigating through cluttered environments in minimum time.

Third, we investigated learning deep sensorimotor policies for vision-based agile flight. Our solution [12], [13] combines visual representation learning to extract feature representations and privileged imitation learning to train a vision-based policy. To enhance the robustness of the vision-based policy, it is crucial to have consistent feature representations, which can be obtained by utilizing contrastive learning along with data augmentation.

In conclusion, our findings underscore the pivotal role of reinforcement learning in empowering drones to achieve their maximum potential while remaining resilient against disturbances.

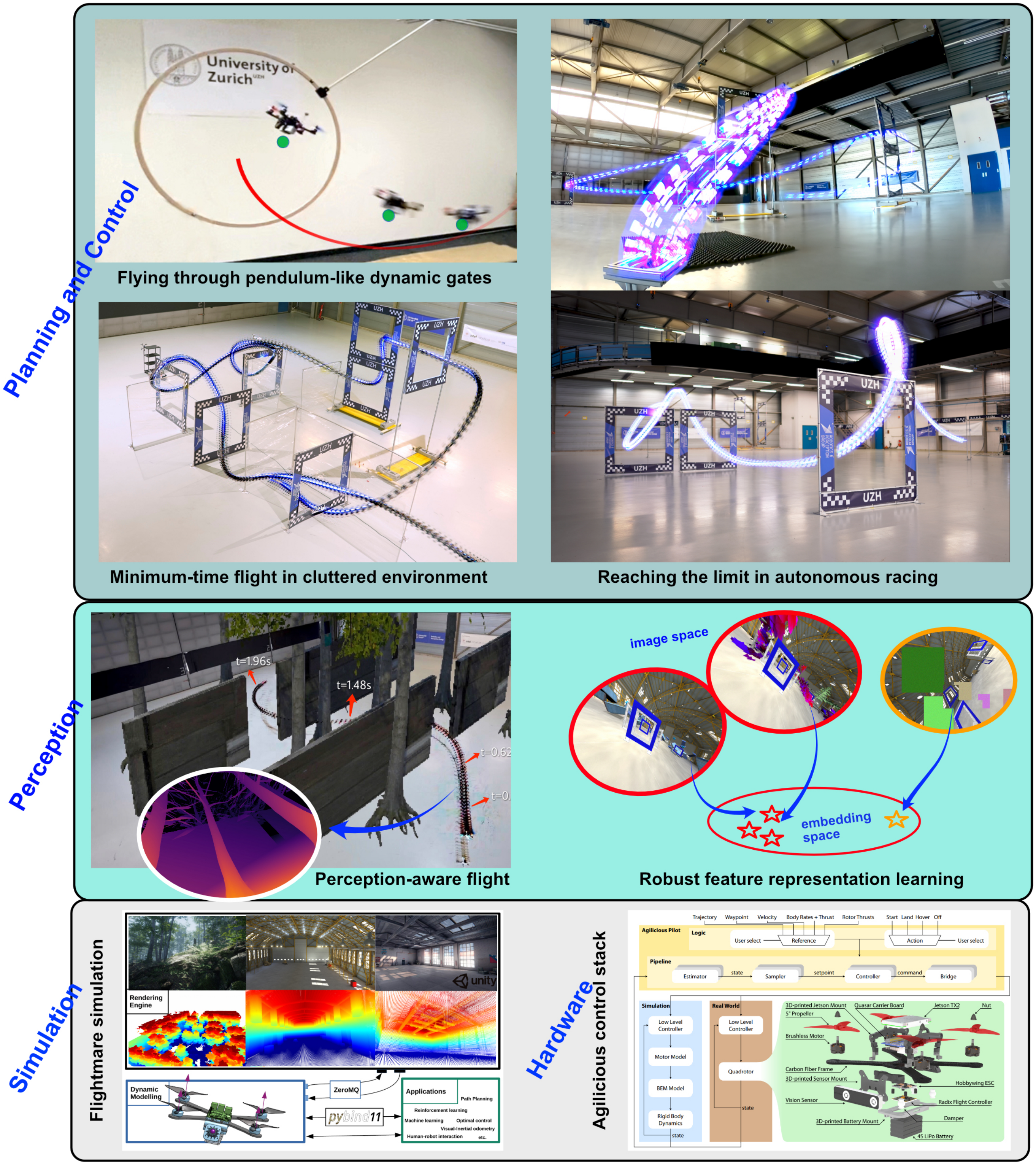

Fig. 1: Learning for agile flight: a synergy of simulation, model predictive control, reinforcement learning, and visual representation learning.

III. Results

A. Flying Through Dynamic Gates

In [7], [8], we present a *policy-search-for-model-predictive-control* framework for agile drone flight in dynamic environments. Flying through fast-moving gates is a proxy task to develop autonomous systems that can navigate the vehicle through rapidly changing environments. However, this is a challenging task as it requires planning an accurate trajectory that passes through the center of moving gates while also controlling the quadrotor to follow the trajectory precisely.

We utilize model predictive control (MPC) as a parameterized controller and formulate the search for high-level decision variables for MPC as a probabilistic policy search problem. A key advantage of our approach over the standard MPC formulation is that the desired traversal time, which is difficult to optimize simultaneously with other state variables, can be learned offline and selected adaptively at runtime.

The resulting controller consists of a high-level neural network policy and an MPC. The high-level policy was used for adaptively making high-level decisions for the MPC. A visualization of the framework is given in Fig 2. Given the predicted decision variables, MPC solves an optimization problem and generates control commands for the vehicle. Our controller achieved robust and real-time control performance in both simulation and the real world. A demonstration of the real-world experiment is shown in Fig. 1, where we demonstrated an agile drone flying through a pendulum-like dynamic gate.

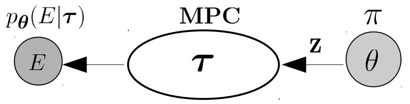

Fig. 2: A graphical model of policy search for model predictive control.

B. Reaching the Limit in Autonomous Racing

The task of autonomous drone racing is to fly a quadrotor through a sequence of gates in a given order in minimum time. Solving this problem requires algorithms to be efficient and fast. Suboptimal control policies readily manifest themselves in reduced task performance, making drone racing a particularly demanding and instructive setting for testing the limits of control design paradigms.

In [9], [10], we leverage deep reinforcement learning and parallel simulation to learn extremely aggressive trajectories that are close to their time-optimal solutions. This is the first learning-based approach for tackling time-optimal quadrotor flight using reinforcement learning. Our method exhibits computational advantages over approaches based on trajectory optimization.

Furthermore, we push an agile drone to its maximum performance, achieving a peak acceleration greater than 12g and a peak velocity of 108 $\mathrm{km\,h^{-1}}$. Our policy achieves superhuman control within minutes of training on a standard workstation. Fig 1 displays time-lapse illustrations of the racing drone controlled by our RL policy in an indoor flying arena.

C. Optimal Control versus Reinforcement Learning

Some of the most impressive achievements of RL are beyond the reach of existing optimal control (OC) systems. However, most of these successes are empirical. Less attention has been paid to the systematic study of fundamental factors that have led to the success of RL or have limited OC. We argue that this question can be investigated along two axes: the optimization method and the optimization objective. On one hand, RL and OC can be viewed as two different optimization methods and we can ask which method can achieve a more robust solution given the same cost function. On the other hand, given that RL and OC address a given robot control problem by optimizing different objectives, we can ask which optimization objective can lead to more robust task performance.

In [9], our main contribution is the study of reinforcement learning and optimal control from the fundamental perspective of the *optimization method and optimization objective*. Our results indicate that RL does not outperform OC because RL optimizes its objective better. Rather, RL outperforms OC because it optimizes a better objective. Specifically, RL directly maximizes a task-level objective, which leads to more robust control performance in the presence of unmodeled dynamics. In the drone racing context, RL can optimize a highly nonlinear and nonconvex gate-progress reward directly, removing the need for a reference time trajectory or a continuous 3D path. In contrast, OC is limited by its decomposition of the problem into planning and control, which requires an intermediate representation in the form of a trajectory or path, thus limiting the range of control policies that can be expressed by the system. In addition, RL can leverage domain randomization to achieve extra robustness and avoid overfitting, where the agent is trained on a variety of simulated environments with varying settings.

D. Minimum-time Flight in Cluttered Environments

Minimum-time flight in cluttered environments is an important problem since it opens up possibilities for many time-critical applications in the real world, such as research and rescue. In [11], we leverage reinforcement learning and topological path planning to train robust neural network controllers for minimum-time quadrotor flight in cluttered environments. The key ingredients of our approach to minimum-time flight in cluttered environments are three-fold: 1) generation of a topological guiding path using a probabilistic roadmap, 2) a novel task formulation that combines progress maximization along the guiding path with obstacle avoidance, and 3) combining curriculum training with deep reinforcement learning.

We show that the presented method can achieve a high success rate in flying minimum-time policies in cluttered environments and outperforms classical approaches that rely on traditional planning and control. The policy is trained entirely in simulation and then transferred to the real world without fine-tuning, with a peak flight speed of 42 $\mathrm{km\,h^{-1}}$ and a maximum accelerations of 3.6g.

E. Perception-aware Flight in Cluttered Environments

In [12], we tackle the problem of vision-based, minimum-time flight in cluttered, known environments for quadrotors. Minimum-time flight requires the vehicle to operate on the edge of both its physical limits and its perceptual limits (e.g., limited field of view). The limited field of view of the onboard camera is particularly constraining for quadrotors due to their underactuated nature: in the most common configuration, all the rotors point in the same direction, which causes the robot to accelerate only in this direction. If the camera is rigidly attached to the drone, this means that a trade-off must be found between maximizing flight performance and optimizing the visibility of regions of interest.

We propose a vision-based navigation system to fly a quadrotor through cluttered environments at high-speed with perception awareness. Our method combines imitation learning and reinforcement learning (RL) by leveraging a privileged learning-by-cheating framework. We begin by training a state-based teacher policy using deep RL to fly a minimum-time trajectory in cluttered environments. This policy integrates progress maximization and obstacle avoidance with a perception-aware reward that aligns the camera orientation with the flight direction. Next, by imitating the teacher policy, we train a vision-based policy that does not rely on privileged information about the obstacles. The resulting vision-based policy achieves high-speed flight and high success rates. We show that our policy has very low computational latency (just 1.4 ms) compared to classical methods with intermediate map representations that have 10 times higher latency.

We test the closed-loop control performance of our vision-based policy in the real world using hardware-in-the-loop (HITL) simulation and the agilicious control stack [6]. HITL involves flying a physical quadrotor in a motion-capture system while observing virtual photorealistic environments that are updated in real-time.

F. Robust Feature Representation Learning

Scene transfer in mobile robotics is a highly relevant and challenging problem as a robot should be able to perform a task outside of a known environment in the real world. We study the scene transfer problem in the context of autonomous vision-based drone racing. Vision-based autonomous drone racing is a challenging navigation task that demands the vehicle to operate at the edge of the vehicle limits. High speeds and quick rotations of the camera induce motion blur and rapid illumination changes, adding to the difficulty of the task.

In [13], we leverage contrastive learning and data augmentation to train a perception network that can extract useful feature representations from high-dimensional images. Using contrastive learning, the perception network learns to focus on useful visual features while ignoring irrelevant backgrounds. In the drone racing context, the gate is the most relevant visual information for the task. Thus, in the feature space, representations are closely clustered when the robot is at the same place and widely dispersed when the robot visits different locations.

Given obtained feature representations, a vision-based control policy is trained using a privileged learning-by-cheating framework. Our experiments, conducted in our realistic simulator [5], show that our vision-based deep sensorimotor policy achieves the same level of racing performance as state-based policies while being resilient against unseen visual disturbances and distractors.

IV. Discussion and Conclusion

While our results are promising, it's crucial to point out that real-world deployments will always present unpredicted challenges. The success of RL over OC in our tasks doesn't necessarily diminish the relevance of OC in drone control nor in other robotic applications. It is also worth noting that while we achieved significant milestones in drone racing, genuine real-world settings, including varied conditions and dynamic environments, demand further adaptability.

Future endeavors might explore ways to investigate visual representation learning further, possibly incorporating online adaptation to dynamic environments, and bridge the gap between simulation and the real world. To accelerate research and let researchers focus on their problems, we have made our simulator Flightmare [5] and our control stack Agilicious [6] open-source.

References

[1] X. Zhou, X. Wen, Z. Wang, Y. Gao, H. Li, Q. Wang, T. Yang, H. Lu, Y. Cao, C. Xu, *et al.*, "Swarm of micro flying robots in the wild," *Science Robotics*, vol. 7, no. 66, p. eabm5954, 2022.

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

[3] J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter, "Learning agile and dynamic motor skills for legged robots," *Science Robotics*, vol. 4, no. 26, 2019.

[4] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Science robotics*, vol. 5, no. 47, 2020.

[5] Y. Song, S. Naji, E. Kaufmann, A. Loquercio, and D. Scaramuzza, "Flightmare: A flexible quadrotor simulator," in *Conference on Robot Learning*, 2020.

[6] P. Foehn, E. Kaufmann, A. Romero, R. Penicka, S. Sun, L. Bauersfeld, T. Laengle, G. Cioffi, Y. Song, A. Loquercio, and D. Scaramuzza, "Agilicious: Open-source and open-hardware agile quadrotor for vision-based flight," *Science Robotics*, vol. 7, no. 67, p. eabl6259, 2022. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.abl6259

[7] Y. Song and D. Scaramuzza, "Policy search for model predictive control with application to agile drone flight," *IEEE Transactions on Robotics*, pp. 1–17, 2022.

[8] ——, "Learning high-level policies for model predictive control," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[9] Y. Song, A. Romero, M. Müller, V. Koltun, and D. Scaramuzza, "Reaching the limit in autonomous racing: Optimal control versus reinforcement learning," *Science Robotics*, vol. 8, no. 82, p. eadg1462, 2023.

[10] Y. Song, M. Steinweg, E. Kaufmann, and D. Scaramuzza, "Autonomous drone racing with deep reinforcement learning," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2021.

[11] R. Penicka, Y. Song, E. Kaufmann, and D. Scaramuzza, "Learning minimum-time flight in cluttered environments," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7209–7216, 2022.

[12] Y. Song, K. Shi, R. Penicka, and D. Scaramuzza, "Learning perception-aware agile flight in cluttered environments," in *IEEE International Conference on Robotics and Automation, ICRA 2023, London*. IEEE, 2023, pp. 1989–1995.

[13] J. Fu, Y. Song, Y. Wu, F. Yu, and D. Scaramuzza, "Learning deep sensorimotor policies for vision-based autonomous drone racing," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2023.