Superhuman Safe and Agile Racing through Multi-Agent Reinforcement Learning

Ismail Geles[1], Leonard Bauersfeld[1], Markus Wulfmeier[2], Davide Scaramuzza[1]

***Abstract*— Autonomous systems have achieved superhuman performance in isolation, yet remain brittle in shared, dynamic real-world spaces. This failure stems from the dominant single-agent paradigm, where other actors are ignored or treated as environmental noise. We show that multi-agent reinforcement learning provides the essential safety scaffolding required for real-world interaction. Using high-speed quadrotor racing as a testbed, we train agents through league-based self-play to navigate aerodynamic interactions and strategic maneuvering with a variable number of competitors. Our agents outperform a champion-level human pilot in multi-player races at speeds exceeding 22 m/s, while reducing collision rates by 50% compared to single-agent baselines. Crucially, training with diverse artificial opponents enables zero-shot generalization to safer human interaction. These results suggest that the path to robust robotic co-existence lies not in isolated safety constraints, but in the rigorous demands of multi-agent interaction. Multimedia materials are available at: https://rpg.ifi.uzh.ch/marl**

I. Introduction

Autonomous robots are increasingly operating in the physical world. Learned policies now enable walking robots to traverse challenging terrain [1], [2], mobile manipulators to perform household tasks [3], and aerial vehicles to navigate complex environments at high speed [4]–[8]. Yet, a striking asymmetry persists: while single-agent systems have matured from laboratory demonstrations to practical deployment, multi-agent coordination remains largely confined to simulation [9], even in highly realistic and competitive domains such as autonomous car racing [10]. This gap presents a fundamental barrier to scaling autonomous systems, as many critical applications, from warehouse logistics to search and rescue to urban air mobility, inherently require multiple robots operating in shared physical space alongside humans. Reinforcement learning has achieved superhuman performance across increasingly complex domains, from Go [11], [12] to the vast strategy spaces of StarCraft II [13] and Dota 2 [14]. A key insight from these systems is that training against a diverse population of opponents prevents strategic overfitting and produces generalizable behaviors [15], [16]. Recent work has begun to bridge the gap to physical systems, demonstrating competitive behaviors in robot soccer [17], [18], small-scale car racing [19], and two-player drone racing [20]. However, these systems either operate at low speeds and limited safety challenges, or are restricted to two-player interactions that do not capture the combinatorial complexity of multi-agent coordination. Whether multi-agent learning can enable safe coordination at speeds where collisions are catastrophic, and reaction margins are measured in milliseconds, remains an open problem.

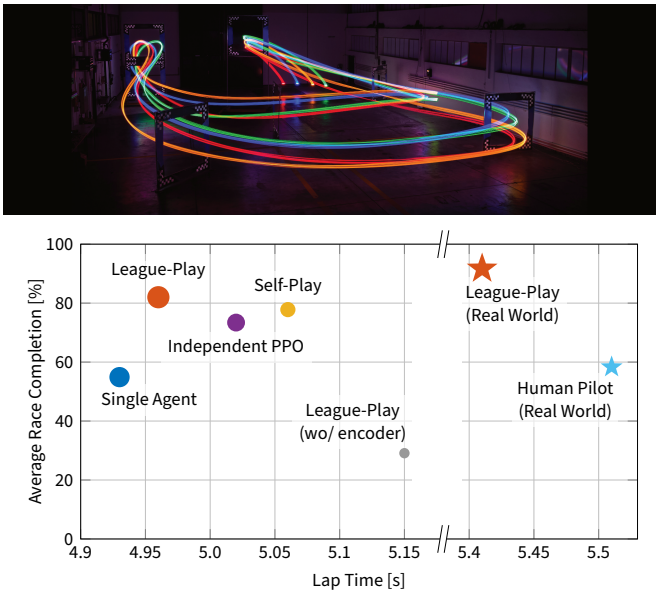

Fig. 1: Top: Real-world deployment with four autonomous agents at speeds exceeding 22 m/s on the Split-S racetrack. Bottom: Evaluation of over 64,000 simulated four-player races comparing lap time against race completion. Stars indicate real-world deployment results.

Multi-player quadrotor racing provides a uniquely demanding testbed for this question. The domain demands performance at the physical limits of vehicle dynamics, with speeds exceeding 100 km/h and accelerations surpassing 12 g [6]. Prior autonomous racing systems have reached these limits but optimize purely for lap time regardless of competitive context [5]–[7]. When multiple vehicles share the track at high speeds, the problem changes significantly. Gates become contested narrow passages, aerodynamic downwash from neighboring vehicles perturbs flight dynamics, and every trajectory constrains every other. The challenge shifts from pure speed to a negotiation between performance and survival.

Here we present a multi-agent reinforcement learning framework that addresses this challenge, achieving superhuman safe multi-player quadrotor racing at speeds reaching 22 m/s and accelerations of up to 7 g. Central to our approach is league-play, inspired by league training [13] and fictitious self-play [21]. Agents train against a diverse, evolving population of competitors drawn from past training

[1] These authors are with the Robotics and Perception Group, Department of Informatics, University of Zurich, Switzerland.

[2] Markus Wulfmeier is with Google DeepMind, London, UK.

This work was supported by the European Union's Horizon Europe Research and Innovation Programme under grant agreement No. 101120732 (AUTOASSESS) and the European Research Council (ERC) under grant agreement No. 864042 (AGILEFLIGHT).

checkpoints and from policies obtained through alternative training regimes. A permutation-invariant attention encoder based on Perceiver [22] allows each agent to process observations invariant to number and order of competitors, while a particle-based downwash model captures aerodynamic interactions in close-proximity flight. We validate the framework in real-world races with up to four competitors, including a champion-level human pilot. Compared to single-agent baselines, league-play training reduces collisions by 50% while preserving competitive lap times. Policies transfer zero-shot to races against human expert pilots, achieving over 90% race completion rates.

II. Methodology

A. Problem Formulation

We formulate multi-agent racing as a Markov game [23], the multi-agent extension of the Markov decision process. The environment is shared between N agents, where at every time-step t each agent i observes its own state $s_{t,\text{ego}}$ and the states of other agents $s_{t,\text{other}} = \{s^{(j)}_{t,\text{other}}\}_{j \neq i}$. Each agent seeks to maximize its expected cumulative reward given its own policy π_{ego} and the policies of other agents π_{other}:

$$J(\pi_{\text{ego}}) = \mathbb{E}_{\pi_{\text{ego}}, \pi_{\text{other}}} \left[\sum_{t=0}^{T} \gamma^t r_t \mid s_{t,\text{ego}}, s_{t,\text{other}} \right] \tag{1}$$

where γ is the discount factor and r_t is the reward at time-step t. Unlike single-agent settings, the optimal policy depends on the behavior of other agents, motivating training against diverse opponents.

The ego state $s_{t,\text{ego}}$ comprises the agent's position, velocity, orientation, and gate-relative features following [6]. Opponent observations for each competitor j are defined as:

$$s^{(j)}_{t,\text{other}} = \left[\mathbf{p}^{(j)}_{\text{rel}}, \mathbf{v}^{(j)}_{\text{rel}} \right] \tag{2}$$

where $\mathbf{p}^{(j)}_{\text{rel}} \in \mathbb{R}^3$ and $\mathbf{v}^{(j)}_{\text{rel}} \in \mathbb{R}^3$ denote the relative position and velocity of opponent j with respect to the ego agent. The action space consists of collective thrust and body rate commands $\mathbf{a}_t = [c, \omega_x, \omega_y, \omega_z] \in \mathbb{R}^4$, tracked by a low-level controller [5], [6].

B. Policy Architecture

To handle variable numbers of competitors, opponent observations $\{s^{(j)}_{t,\text{other}}\}_{j=1}^{N-1}$ are processed through a Perceiver-based attention encoder [22]. The encoder uses 4 learned latent queries, each attending to the full set of opponent observations via 4-head cross-attention with a head dimension of 32, producing a fixed-dimensional, permutation-invariant representation regardless of the number of competitors. The encoding is concatenated with the ego state and passed to recurrent actor and critic networks. Both networks comprise a single LSTM layer with 256 hidden units followed by two fully connected layers with 512 units each. The recurrent architecture enables agents to maintain beliefs about opponent intentions and race dynamics across time steps, capturing temporal dependencies that are critical for anticipatory behavior.

C. League-Play Training

Policies are trained using a recurrent variant of Proximal Policy Optimization (PPO) [24] for 200 million environment interactions. Training follows a league-play paradigm inspired by [13]. In each training iteration, a single learning agent is paired with opponents sampled from two distinct pools: historical checkpoints from its own training (fictitious self-play) and a fixed league of diverse opponents. With probability 75%, opponents are drawn from the agent's own checkpoint history; otherwise, opponents are sampled from the league pool. Checkpoint selection is biased toward more recent snapshots, which exhibit stronger performance, while maintaining non-zero probability of encountering earlier strategies to prevent catastrophic forgetting.

The league pool contains 20 fixed policies capturing diverse competitive behaviors: four single-agent policies optimized purely for lap time without opponent awareness, and sixteen independently trained PPO policies drawn from jointly trained agents at various stages of training. This diversity exposes the learning agent to competitive scenarios absent from self-play alone, where opponents share similar strategic tendencies.

D. Reward and Training Setup

We employ a dense reward function that extends prior work in learned quadrotor control [6] with terms for multi-agent interaction. The per-step reward combines four components: a progress term incentivizing advancement toward the next gate, a body rate penalty promoting smooth flight, a ranking reward encouraging competitive behavior by rewarding relative race position, and a proximity penalty that provides a continuous shaping signal discouraging unsafe approaches to opponents. Collisions with walls, gates, or other agents terminate the episode with a penalty proportional to impact severity. To account for scenarios where vehicles make brief contact but remain airworthy, 10% of inter-agent collisions are randomly selected to be non-terminal.

Training is conducted in simulation using Flightmare [25] with the Agilicious framework [26], parallelized across 144 environments each simulating four quadrotors. Aerodynamic interactions between agents are modeled through a particle-based downwash simulation that captures thrust disturbances from nearby competitors. Agents flying through opponent downwash experience altered aerodynamic forces that require compensatory control responses, an interaction unique to aerial multi-agent systems. A staged curriculum progressively introduces opponent visibility and physical interactions, allowing agents to first acquire proficient single-agent racing behavior before learning multi-agent coordination. Domain randomization across vehicle dynamics and initial conditions enables zero-shot sim-to-real transfer [5], [6].

III. Experiments

We evaluate our approach in both real-world and simulated multi-agent settings using a single policy trained with four agents that generalizes to varying numbers of opponents.

TABLE I: Real-world racing performance. Comparison between league-play policy (LP) and expert human pilot M. Schaepper. Race completion measures the fraction of the three-lap race completed before collision.

Config.	Pilot	Trials	Avg. Race Compl. [%]	Best Lap [s]	Best Race [s]
Time Trial	LP (ours)	13	100.00	**5.540**	**16.065**
	M. Schaepper	10	100.00	6.627	17.109
AI Only (4)	LP (ours)	12	90.87	6.079	16.811
AI vs Human (3v1)	LP (ours)	12	**91.67**	**5.998**	**16.639**
	M. Schaepper	4	58.33	6.554	25.776

A. Experimental Setup

Real-world experiments are conducted in an indoor flight arena equipped with a 42-camera motion capture system. The racetrack comprises seven gates arranged in an approximately 75-meter circuit, including a Split-S maneuver requiring sequential passage through vertically stacked gates. All quadrotors are identical 220-gram racing platforms based on the Agilicious framework [26], with a thrust-to-weight ratio of 6.5. Autonomous agents execute learned policies at 50 Hz using motion-capture state estimation.

B. Real-World Multi-Player Racing

Physical deployment is validated through racing experiments spanning time trials, AI-only races, and mixed human-AI competitions against Marvin Schaepper, five-time Swiss national drone racing champion. The human pilot operates using standard first-person-view equipment, while autonomous agents observe their own state alongside the relative positions and velocities of all competitors. Average race completion serves as the primary safety metric, measuring the fraction of the three-lap race (21 gates) completed before termination due to collision. Table I summarizes performance across experimental configurations.

In time trials, both the human pilot and our policy achieved 100% race completion, with our policy recording a fastest lap of 5.540 s compared to the human champion's 6.627 s. The benefits of interaction-aware training become apparent under multi-agent competition. In the configuration of three autonomous agents racing alongside the human pilot, the learned policies sustained 91.67% completion compared to 58.33% for the human, despite the increased density of nearby competitors. Across all configurations, the league-play policy finished ahead of the human pilot in nearly every race. Analysis of race trajectories reveals a consistent pattern: the human pilot, typically trailing the autonomous agents, attempted increasingly aggressive maneuvers to close the gap, often resulting in gate collisions or loss of control.

The learned policies, by contrast, maintained consistent safety margins regardless of competitive standing, with anticipatory behaviors emerging through competitive self-play without explicit programming: agents learn to yield when overtaking is unsafe and exploit opportunities created by competitor positioning.

C. Large-Scale Simulated Evaluation

To evaluate generalization beyond the training distribution, we conducted large-scale simulated competitions across 64,000 four-player races against diverse opponent behaviors, including strategies never encountered during training. Five training strategies are compared: (i) single-agent policies without opponent observations, (ii) independent PPO with opponent observations, (iii) fictitious self-play, (iv) league-play, and (v) league-play without the Perceiver-based attention encoder, see Figure 1.

A clear trade-off between speed and safety emerges across training paradigms. Single-agent policies achieve the fastest average lap time of 4.93 s but exhibit the highest crash rates, failing to complete the majority of races. League-play achieves the highest race completion among all methods while maintaining an average lap time of 4.96 s, only 0.03 s slower than the single-agent baseline. The ablation without the Perceiver encoder exhibits degraded performance below single-agent baselines, confirming that the attention-based encoder is essential for safe multi-agent coordination.

Crash rate analysis reveals that opponent collisions dominate across all methods, with gate and wall collisions contributing only a small fraction of total failures, underscoring that dynamic competitors pose a far greater safety challenge than static obstacles.

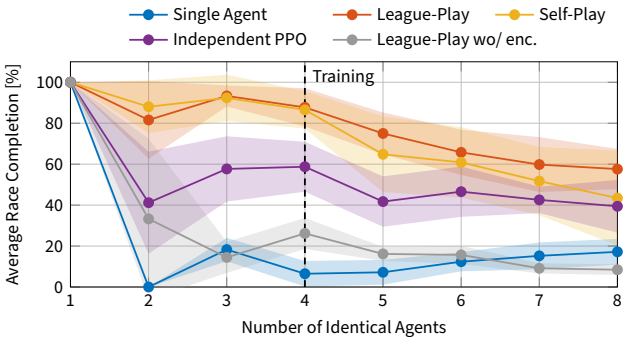

Fig. 2: Average race completion through self-evaluation with identical policies from solo to 8-player races. Each data point represents four policies per method, each completing 64 races with varied starting positions. Shaded regions indicate the average of standard deviations per policy.

D. Scalability Beyond Training Conditions

To test generalization beyond the four-agent training configuration, we conducted self-evaluation experiments with progressively increasing competitor counts from solo to 8-player races (Figure 2). Single-agent policies exhibit severe degradation, selecting identical racing lines and resulting in 85% collision probability at the first gate in four-player configurations. Independent PPO agents achieve near-perfect

avoidance among co-trained partners but drop to approximately 60% race completion in self-evaluation, indicating overfitting to specific training partners. League-play sustains higher completion rates as competitor count increases, gradually decreasing to 56% at eight agents. Averaged across all agent counts, league-play reduces the average crash rate by over 50% relative to single-agent policies (23.30% versus 77.99%).

IV. Conclusion

Multi-agent reinforcement learning, grounded in diverse self-play and physical interaction modeling, achieves safe high-speed coordination in shared environments. A central finding is that interaction-aware training produces policies that maintain consistent safety margins regardless of competitive standing, unlike human pilots who adopt increasingly aggressive strategies when trailing. The ability to internalize such behavior through self-play, without explicit safety constraints, suggests a path toward autonomous systems that are safe by design rather than by restriction.

While our system currently relies on motion-capture state estimation rather than onboard vision [8], and all experiments use a single track layout, the core insight that training in diverse multi-agent settings produces generalizable and safe interaction strategies extends naturally beyond racing to domains such as warehouse logistics, urban air mobility, and search and rescue, where multiple autonomous agents must operate safely alongside humans.

References

[1] T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning robust perceptive locomotion for quadrupedal robots in the wild," *Science Robotics*, vol. 7, no. 62, p. eabk2822, 2022. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.abk2822

[2] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Science Robotics*, vol. 5, no. 47, p. eabc5986, 2020. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.abc5986

[3] Z. Fu, T. Z. Zhao, and C. Finn, "Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation," *Conference on Robot Learning (CoRL)*, 2024.

[4] A. Loquercio, E. Kaufmann, R. Ranftl, M. Müller, V. Koltun, and D. Scaramuzza, "Learning high-speed flight in the wild," *Science Robotics*, vol. 6, no. 59, p. eabg5810, 2021. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.abg5810

[5] E. Kaufmann, L. Bauersfeld, A. Loquercio, M. Müller, V. Koltun, and D. Scaramuzza, "Champion-level drone racing using deep reinforcement learning," *Nature*, vol. 620, no. 7976, pp. 982–987, 2023. [Online]. Available: https://doi.org/10.1038/s41586-023-06419-4

[6] 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. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.adg1462

[7] S. A. Bahnam, R. Ferede, T. M. Blaha, A. E. Lang, E. Lucassen, Q. Missinne, A. E. Verraest, C. De Wagter, and G. C. de Croon, "Monorace: Winning champion-level drone racing with robust monocular ai," *arXiv preprint arXiv:2601.15222*, 2026.

[8] I. Geles, L. Bauersfeld, A. Romero, J. Xing, and D. Scaramuzza, "Demonstrating Agile Flight from Pixels without State Estimation," *Proceedings of Robotics: Science and Systems*, July 2024.

[9] S. Gronauer and K. Diepold, "Multi-agent deep reinforcement learning: a survey," *Artificial Intelligence Review*, vol. 55, no. 2, pp. 895–943, 2022.

[10] P. R. Wurman, S. Barrett, K. Kawamoto, J. MacGlashan, K. Subramanian, T. J. Walsh, R. Capobianco, A. Devlic, F. Eckert, F. Fuchs, L. Gilpin, P. Khandelwal, V. Kompella, H. Lin, P. MacAlpine, D. Oller, T. Seno, C. Sherstan, M. D. Thomure, H. Aghabozorgi, L. Barrett, R. Douglas, D. Whitehead, P. Dürr, P. Stone, M. Spranger, and H. Kitano, "Outracing champion gran turismo drivers with deep reinforcement learning," *Nature*, vol. 602, no. 7896, pp. 223–228, 2022. [Online]. Available: https://doi.org/10.1038/s41586-021-04357-7

[11] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[12] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, and A. Bolton, "Mastering the game of Go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[13] O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, and P. Georgiev, "Grandmaster level in StarCraft II using multi-agent reinforcement learning," *Nature*, vol. 575, no. 7782, pp. 350–354, 2019.

[14] C. Berner, G. Brockman, B. Chan, V. Cheung, P. Debiak, C. Dennison, D. Farhi, Q. Fischer, S. Hashme, C. Hesse, R. Jozefowicz, S. Gray, C. Olsson, J. Pachocki, M. Petrov, H. P. d.O. Pinto, J. Raiman, T. Salimans, J. Schlatter, J. Schneider, S. Sidor, I. Sutskever, J. Tang, F. Wolski, and S. Zhang, "Dota 2 with large scale deep reinforcement learning," *arXiv preprint arXiv:1912.06680*, 2019.

[15] M. Lanctot, V. Zambaldi, A. Gruslys, A. Lazaridou, K. Tuyls, J. Pérolat, D. Silver, and T. Graepel, "A unified game-theoretic approach to multiagent reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[16] D. Balduzzi, M. Garnelo, Y. Bachrach, W. Czarnecki, J. Perolat, M. Jaderberg, and T. Graepel, "Open-ended learning in symmetric zero-sum games," *International Conference on Machine Learning*, pp. 434–443, 2019.

[17] T. Haarnoja, B. Moran, G. Lever, S. H. Huang, D. Tirumala, J. Humplik, M. Wulfmeier, S. Tunyasuvunakool, N. Y. Siegel, R. Hafner *et al.*, "Learning agile soccer skills for a bipedal robot with deep reinforcement learning," *Science Robotics*, vol. 9, no. 89, p. eadi8022, 2024.

[18] D. Tirumala, M. Wulfmeier, B. Moran, S. Huang, J. Humplik, G. Lever, T. Haarnoja, L. Hasenclever, A. Byravan, N. Batchelor, N. sreendra, K. Patel, M. Gwira, F. Nori, M. Riedmiller, and N. Heess, "Learning robot soccer from egocentric vision with deep reinforcement learning," *8th Annual Conference on Robot Learning*, 2024.

[19] P. Werner, T. Seyde, P. Drews, T. M. Balch, I. Gilitschenski, W. Schwarting, G. Rosman, S. Karaman, and D. Rus, "Dynamic multi-team racing: Competitive driving on 1/10-th scale vehicles via learning in simulation," *7th Annual Conference on Robot Learning*, 2023.

[20] V. Pasumarti, L. Bianchi, and A. Loquercio, "Agile flight emerges from multi-agent competitive racing," *2026 IEEE International Conference on Robotics and Automation (ICRA)*, 2026.

[21] J. Heinrich, M. Lanctot, and D. Silver, "Fictitious self-play in extensive-form games," *International Conference on Machine Learning*, pp. 805–813, 2015.

[22] A. Jaegle, F. Gimeno, A. Brock, O. Vinyals, A. Zisserman, and J. Carreira, "Perceiver: General perception with iterative attention," *International Conference on Machine Learning*, pp. 4651–4664, 2021.

[23] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Machine Learning Proceedings 1994*, W. W. Cohen and H. Hirsh, Eds. San Francisco (CA): Morgan Kaufmann, 1994, pp. 157–163. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9781558603356500271

[24] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[25] Y. Song, S. Naji, E. Kaufmann, A. Loquercio, and D. Scaramuzza, "Flightmare: A flexible quadrotor simulator," *Proceedings of the 2020 Conference on Robot Learning*, pp. 1147–1157, 2021.

[26] 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.