RVC-NMPC: Mutual Collision Avoidance for Agile UAV Flight via Nonlinear Model Predictive Control with Reciprocal Velocity Constraints

Vít Krátký[1⋆], Robert Pěnička[1], Parakh Manoj Gupta[1], and Martin Saska[1]

Supplementary material

Video: `https://youtu.be/cIWm7xg0upc`

I. Extended Abstract

The deployment of Unmanned Aerial Vehicles (UAVs) in the past decade was mostly limited to single-robot applications in an isolated operational space. However, in recent years the development targets applications leading to numerous UAVs operating in an open-air space shared with other parties of air traffic (e.g., package delivery, area monitoring, and precision agriculture). This brings to the forefront the problem of mutual collision avoidance, which is one of the key aspects of the safe deployment of robotic systems in real-world applications where robots share an operational space.

Once the UAVs will be deployed on an everyday basis for numerous tasks such as package delivery, they are expected to operate in much more dense environments than, e.g., aeroplanes because of their limited flight altitudes and higher density of starting and delivery locations. This makes relying on centralized planning and scheduling impractical. In the realm of an efficient operation of UAVs, methods that allow collision avoidance in a high-speed agile flight are of particular interest because they allow to fully exploit the efficiency of UAVs, including the capability of a high-speed agile flight.

The approaches tackling mutual collision avoidance in multi-robot scenarios focus on providing theoretical guarantees but neglect real-world aspects of the problem [1]–[7]. Most of these works assume unrealistic perfect control (reference tracking) [1], they often neglect the kinematic and dynamic constraints of the UAVs [2]–[4] or require knowledge of the future trajectories of all other UAVs which puts high requirements on the communication network bandwidth [5]–[7]. Even with all these unrealistic or highly constraining assumptions, most of these works cannot handle scenarios with velocities exceeding $10\,\mathrm{m\,s^{-1}}$, which is well below the speeds achievable by commercially available drones[3].

In recent years, the major focus in field of multi-robot collision-free trajectory planning was put on development of decentralized methods that require the robots to share not only current state of UAVs, but also their future planned trajectories. Majority of these methods rely on optimization techniques and varies in trajectory parametrization, planning methods, and methods of free space decomposition and obstacle representations [5]–[14]. Some of these works bring contribution by addressing individual aspects of the cooperative navigation problem such as efficient collision resolution in dense environments [11], robustness to communication delays [12], perception- and uncertainty-awareness [13], or deadlock prevention [7]. While these methods show impressive results in cluttered environments, the requirements on knowledge of the other UAVs' states and planned trajectories limits their use to scenarios with cooperating robots sharing the required data and puts high demands on communication network bandwidth. Most of these works also evaluate the results on the level of reference trajectories and do not consider the presence of imperfect controller resulting in deviation from provided reference.

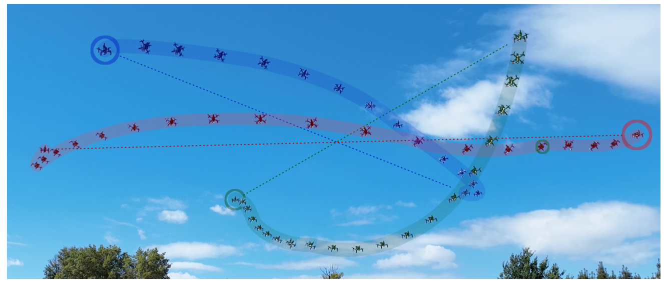

Fig. 1: Deployment of the introduced RVC-NMPC approach in real-world scenario with 3 UAVs navigating to antipodal positions on a circle with radius 10 m.

Reactive approaches for mutual collision avoidance often rely on representing other agents as obstacles with a simplified dynamics (e.g., Velocity Obstacle (VO) [15]). The most direct extensions of VO concept are Reciprocal Velocity Obstacles (RVOs) [16], Optimal Reciprocal Collision Avoidance (ORCA) [2], and V-RVO [17] which improves the efficiency of collision avoidance by letting each agent take half of the responsibility to avoid collision between cooperating agents. While these approaches implement collision avoidance based on position and velocity observations only, they neglect physical constraints of individual platforms and expect immediate change of their velocity.

The lack of consideration of dynamic models is overcome in several adaptations of velocity obstacles, e.g., by using second-order dynamics [18], nonholonomic models [19], and general linear systems [20], [21]. Some approaches overcome the simplicity of these concepts by integrating velocity obstacles or their adaptations with other frameworks such as reinforcement learning [22], [23] or Model Predictive Control (MPC) [1], [24]. However, they still cannot reliably handle simple scenarios with maximum velocities reaching $7\,\mathrm{m\,s^{-1}}$.

[1]Authors are with the Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic.

⋆Corresponding author: `vit.kratky@fel.cvut.cz`

This work was funded by the Czech Science Foundation (GAČR) grant no. 23-06162M.

[3]https://enterprise.dji.com/matrice-30/specs, https://www.skydio.com/x10/

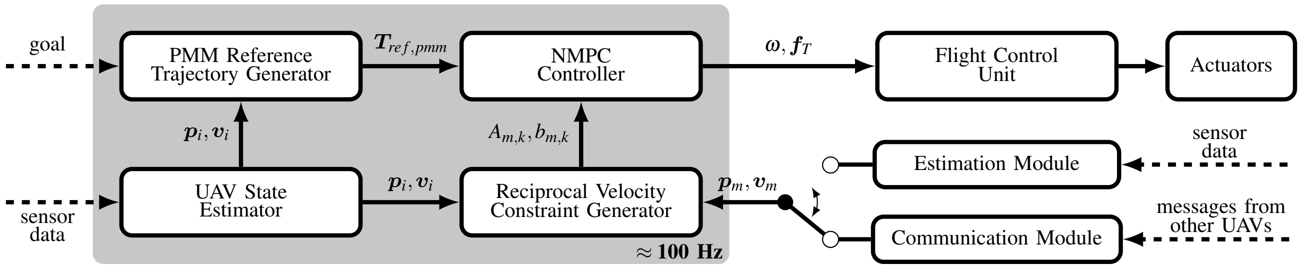

Fig. 2: Block diagram representing a single robot control and navigation pipeline for mutual collision avoidance for agile UAV flight.

Apart from MPC-based approaches, usage of Nonlinear Model Predictive Control (NMPC) in the context of mutual collision avoidance was also already explored in literature [25], [26]. However, while NMPC enables to tackle the nonlinearity of the controlled system and provides more freedom for integration of mutual collision avoidance mechanisms directly into control problem formulation, the increased complexity of the problem makes it computationally demanding which often limits the update period and prediction horizon length, and thus restricts the use of these methods in high-speed scenarios.

To this end, we address the problem of mutual collision avoidance by proposing a novel NMPC-based approach with time-dependent reciprocal velocity constraints that are computed only based on the current position and velocity of the robots. In difference to future trajectories required by state-of-the-art methods, the position and velocity can be obtained by other UAVs through onboard sensor data [27], [28] or through a low-bandwidth communication network, e.g., as part of the Remote Drone ID[4]. Integrating these constraints directly in a control pipeline of the UAVs ensures proper and fast reaction to external disturbances, increases the method's robustness, and allows seamless integration of dynamic constraints. Low computational demands allow all pipeline modules to run at 100 Hz, which further facilitates fast reaction on changes in the behavior of other UAVs and enables efficient use of the method in high-speed scenarios. The proposed approach consists of several modules that process the requested goal destination, sensor data and eventually telemetries of other robots to generate quadrotor control inputs that result in collision-free navigation to the goal destination in multi-robot scenarios. The block diagram of the pipeline is provided in Fig. 2.

The necessary inputs of the designed pipeline include sensor data, which are processed by *UAV State Estimator* providing the estimates of the current robot's position and velocity. This estimate is supplied together with a user-provided goal destination to *PMM Reference Trajectory Generator*, and together with the positions and velocities of other robots, it is also provided to *Reciprocal Velocity Constraint Generator*. The *PMM Reference Trajectory Generator* applies a point-mass model minimum-time trajectory generation approach introduced in [29] for generating a feasible minimum-time trajectory leading from a current state to a goal destination while respecting given kinematic constraints.

The *Reciprocal Velocity Constraint Generator* generates a set of linear reciprocal velocity constraints ensuring mutual collision avoidance among robots based on the positions and velocities of other robots obtained either through *Communication Module* or estimated from the robot's sensor data using an *Estimation Module*. Given the current position and velocity of the robot, the set of velocities for optimal reciprocal collision avoidance [2] is computed for every neighboring robot. The individual set of velocities considering avoidance to other robots are then converted to linear constraints computed in compliance with optimal reciprocal collision avoidance concept [2]. To cope with the latency of data resulting from communication delays and lower frequencies of the incoming messages compared to control loop frequency, the first order linear motion model is applied to predict the current position of other UAVs based on the most recent available information.

Unlike MPC-based approaches applying ORCA [1], [24], the proposed approach misses information about future trajectories, hence it computes velocity constraints only for the current position and velocity and considers this constraint for all transition points. This significantly reduces the computational time without negatively affecting the performance of the method. While applying this constraint on the whole control horizon of duration preserves the required mutual collision avoidance guarantees, such approach is unnecessarily restrictive. Therefore, we introduce the time validity of individual velocity constraints equivalent to time after which the angle between vector representing the relative pose between robots and vector representing relative velocity of robots exceeds $\frac{\pi}{2}$.

The generated reciprocal velocity constraints and the reference trajectory generated by *PMM Reference Trajectory Generator* serve as inputs to *NMPC Controller* generating control inputs that are passed to *Flight Control Unit*, which translates this reference to control commands for individual rotors. The *NMPC Controller* is an adaptation of NMPC control approach [30] enhanced with additional collision avoidance constraints. The complete set of constraints consists of constraints on quadrotor's initial state, constraints corresponding to discretized quadrotor's nonlinear dynamic model adopted from [30], constraints on control inputs, and time-dependent linear velocity constraints for mutual collision avoidance described above. Given the time validity for each linear velocity constraint, these constraints are intro-

[4]https://drone-remote-id.com/

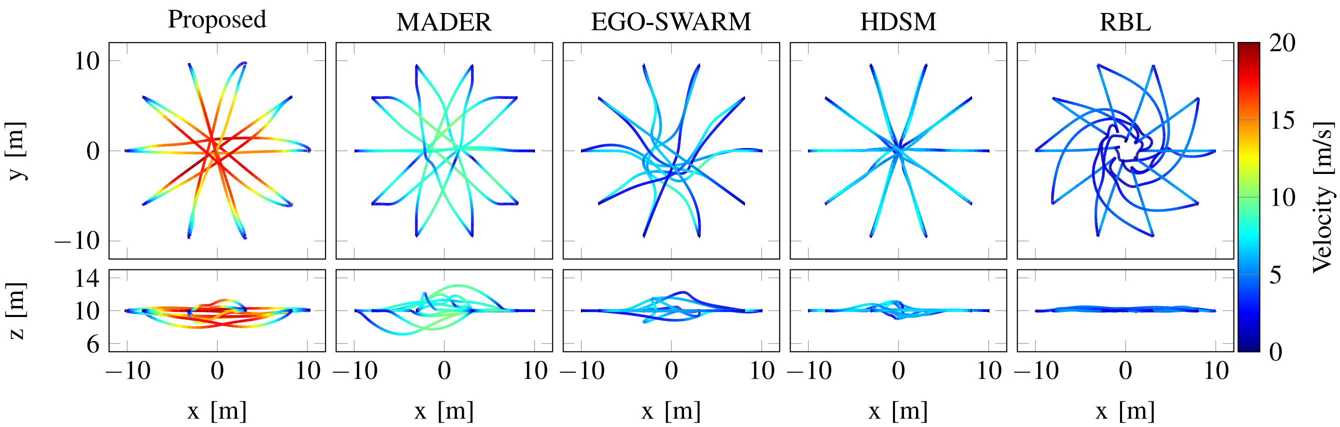

Fig. 3: Qualitative comparison of trajectories generated by individual approaches in scenario involving 10 UAVs navigating to antipodal positions on the circle of diameter 20 m with velocity and acceleration limits $20\,\mathrm{m\,s^{-1}}$ and $40\,\mathrm{m\,s^{-1}}$, respectively.

duced in NMPC formulation as time dependent/transition point dependent variable soft constraints. Such an approach prevents unfeasibility of the defined problem and preserves scalability of the proposed approach.

The proposed RVC-NMPC approach was evaluated through numerous simulations and real-world experiments. Despite the absence of theoretical guarantees, it demonstrated its practicality through collision-free navigation in a three hour long test in an environment with ten robots following random trajectories with velocities and accelerations up to $25\,\mathrm{m\,s^{-1}}$, and $40\,\mathrm{m\,s^{-2}}$, respectively. During this test, 10 UAVs were navigated to more than 50000 goals, travelled total of 6.28×10^5 m with an average velocity $5.8\,\mathrm{m\,s^{-1}}$. The results show that the proposed approach prevents 100 % of violations of minimum mutual distance even in such challenging scenario (see Fig. 4 for detailed results).

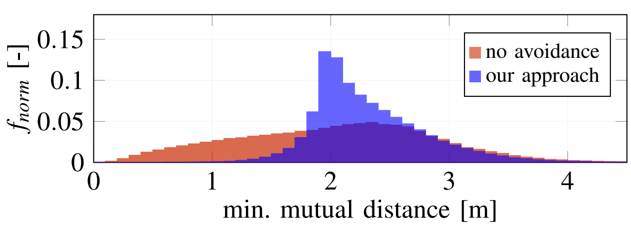

Fig. 4: Histograms of normalized frequencies of minimum mutual distances among UAVs experienced during continuous high-speed navigation in constrained area using the proposed approach (blue) and not applying any collision avoidance mechanism (red).

The performance of the proposed approach was further demonstrated in a scenario involving 10 UAVs simultaneously navigating to antipodal positions on the circle of diameter 20 m (see Fig. 3) with velocity and acceleration constraints $20\,\mathrm{m\,s^{-1}}$ and $40\,\mathrm{m\,s^{-2}}$, respectively. The obtained results show that the proposed approach reduces the flight time by more than 11% compared to state-of-the-art distributed approaches [5], [6], [9], [31] while reaching 100% success rate. The detailed results are presented in Table I.

The robustness of the proposed approach to latency and dropouts in obtained state of other UAVs is demonstrated

TABLE I: Comparison of different approaches for the solution of a simultaneous navigation of 10 UAVs to antipodal positions on a circle of diameter 20 m. The results are averaged over 100 trials.

Approach	Flight time [s]		Vel. $[\mathrm{m\,s^{-1}}]$		Min. dist. [m]	
	mean	std.dev	mean	std.dev	mean	min
MADER [6] - per axis	5.31	0.53	4.75	0.17	0.75	0.38
MADER [6] - norm	5.28	0.35	4.70	0.16	0.72	0.51
EGO-SWARM-2 [9]	8.16	0.50	3.61	0.22	0.67	0.47
HDSM [5] - per axis	3.43	0.17	6.96	0.16	0.67	0.57
HDSM [5] - norm	4.46	0.19	4.78	0.18	0.67	0.56
RBL [31]	11.59	0.74	2.54	0.05	0.87	0.63
Proposed	**3.07**	0.08	**7.36**	0.06	**0.98**	0.81
Proposed with drag	3.84	**0.06**	5.93	**0.05**	**0.98**	**0.82**

through series of simulations with modelled latency, and decrease in incoming frequency of estimated state of other UAVs. While the evaluation is performed in simulation, the modelled errors are emulating real-world conditions resulting from usage of wireless means of communication or estimation of the state of UAVs using onboard sensors and processing. In the following simulations, the collision avoidance radius of individual robots for generation of velocity constraints is set to 2 m with 8 s horizon, and the communication is asynchronous. The detailed results are presented in Fig. 5.

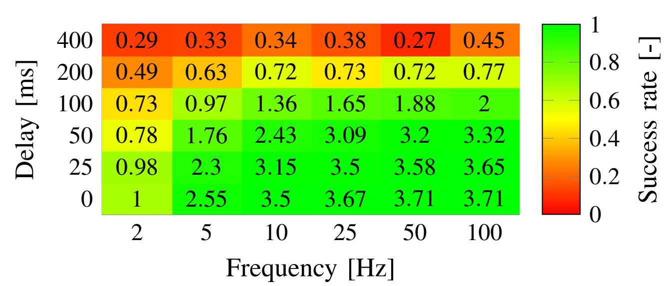

Fig. 5: The success rate (shown in colors of the matrix) and mean minimum mutual distance between UAVs (shown as numbers in the matrix [m]) under varying delay and frequency of messages obtained from other agents. The results for every delay-frequency pair are based on 100 flights in scenario involving 4 UAVs navigating to antipodal positions on the circle of diameter 20 m.

References

[1] H. Cheng, Q. Zhu, Z. Liu, T. Xu, and L. Lin, "Decentralized navigation of multiple agents based on ORCA and model predictive control," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 3446–3451.

[2] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, "Optimal reciprocal collision avoidance for multi-agent navigation," in *IEEE International Conference on Robotics and Automation*, 2010.

[3] A. Levy, C. Keitel, S. Engel, and J. McLurkin, "The Extended Velocity Obstacle and applying ORCA in the real world," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 16–22.

[4] K. Guo, D. Wang, T. Fan, and J. Pan, "VR-ORCA: Variable Responsibility Optimal Reciprocal Collision Avoidance," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4520–4527, 2021.

[5] C. Toumieh and D. Floreano, "High-speed motion planning for aerial swarms in unknown and cluttered environments," *IEEE Transactions on Robotics*, vol. 40, pp. 3642–3656, 2024.

[6] J. Tordesillas and J. P. How, "MADER: Trajectory Planner in Multiagent and Dynamic Environments," *IEEE Transactions on Robotics*, vol. 38, no. 1, pp. 463–476, 2022.

[7] J. Park, Y. Lee, I. Jang, and H. J. Kim, "Dlsc: Distributed multi-agent trajectory planning in maze-like dynamic environments using linear safe corridor," *IEEE Transactions on Robotics*, vol. 39, no. 5, pp. 3739–3758, 2023.

[8] X. Zhou, J. Zhu, H. Zhou, C. Xu, and F. Gao, "Ego-swarm: A fully autonomous and decentralized quadrotor swarm system in cluttered environments," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 4101–4107.

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

[10] X. Zhou, Z. Wang, X. Wen, J. Zhu, C. Xu, and F. Gao, "Decentralized spatial-temporal trajectory planning for multicopter swarms," 2021, arXiv preprint arXiv:2106.12481.

[11] J. Hou, X. Zhou, Z. Gan, and F. Gao, "Enhanced decentralized autonomous aerial robot teams with group planning," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9240–9247, 2022.

[12] K. Kondo, R. Figueroa, J. Rached, J. Tordesillas, P. C. Lusk, and J. P. How, "Robust MADER: Decentralized Multiagent Trajectory Planner Robust to Communication Delay in Dynamic Environments," *IEEE Robotics and Automation Letters*, vol. 9, no. 2, pp. 1476–1483, 2024.

[13] K. Kondo, C. T. Tewari, M. B. Peterson, A. Thomas, J. Kinnari, A. Tagliabue, and J. P. How, "PUMA: Fully Decentralized Uncertainty-aware Multiagent Trajectory Planner with Real-time Image Segmentation-based Frame Alignment," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 13 961–13 967.

[14] V. K. Adajania, S. Zhou, A. K. Singh, and A. P. Schoellig, "AMSwarmX: Safe Swarm Coordination in CompleX Environments via Implicit Non-Convex Decomposition of the Obstacle-Free Space," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 14 555–14 561.

[15] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 1998.

[16] J. van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *2008 IEEE International Conference on Robotics and Automation*, 2008, pp. 1928–1935.

[17] S. H. Arul and D. Manocha, "V-rvo: Decentralized multi-agent collision avoidance using voronoi diagrams and reciprocal velocity obstacles," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 8097–8104.

[18] J. van den Berg, J. Snape, S. J. Guy, and D. Manocha, "Reciprocal collision avoidance with acceleration-velocity obstacles," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 3475–3482.

[19] J. Alonso-Mora, A. Breitenmoser, P. Beardsley, and R. Siegwart, "Reciprocal collision avoidance for multiple car-like robots," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 360–366.

[20] D. Bareiss and J. van den Berg, "Reciprocal collision avoidance for robots with linear dynamics using lqr-obstacles," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 3847–3853.

[21] J. van den Berg, D. Wilkie, S. J. Guy, M. Niethammer, and D. Manocha, "Lqg-obstacles: Feedback control with collision avoidance for mobile robots with motion and sensing uncertainty," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 346–353.

[22] J. Qin, J. Qin, J. Qiu, Q. Liu, M. Li, and Q. Ma, "SRL-ORCA: A Socially Aware Multi-Agent Mapless Navigation Algorithm in Complex Dynamic Scenes," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 143–150, 2024.

[23] Z. Liu, C. Yao, W. Na, C. Liu, and Q. Chen, "MAPPO-Based Optimal Reciprocal Collision Avoidance for Autonomous Mobile Robots in Crowds," in *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2023, pp. 3907–3912.

[24] S. H. Arul and D. Manocha, "DCAD: Decentralized Collision Avoidance With Dynamics Constraints for Agile Quadrotor Swarms," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1191–1198, 2020.

[25] H. Zhu and J. Alonso-Mora, "Chance-Constrained Collision Avoidance for MAVs in Dynamic Environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 776–783, 2019.

[26] M. Goarin, G. Li, A. Saviolo, and G. Loianno, "Decentralized Nonlinear Model Predictive Control for Safe Collision Avoidance in Quadrotor Teams with Limited Detection Range," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 2025, pp. 5387–5393.

[27] Z. Xu, X. Zhan, Y. Xiu, C. Suzuki, and K. Shimada, "Onboard dynamic-object detection and tracking for autonomous robot navigation with rgb-d camera," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 651–658, 2024.

[28] M. Vrba, V. Walter, V. Pritzl, M. Pliska, T. Báča, V. Spurný, D. Heřt, and M. Saska, "On onboard lidar-based flying object detection," *IEEE Transactions on Robotics*, pp. 1–19, 2024.

[29] K. Teissing, M. Novosad, R. Penicka, and M. Saska, "Real-time planning of minimum-time trajectories for agile uav flight," *IEEE Robotics and Automation Letters*, vol. 9, no. 11, pp. 10 351–10 358, 2024.

[30] P. M. Gupta, O. Procházka, J. Hřebec, M. Novosad, R. Pěnička, and M. Saska, "Lol-nmpc: Low-level dynamics integration in nonlinear model predictive control for unmanned aerial vehicles," 2025, arXiv preprint arXiv:2506.02169.

[31] M. Boldrer, A. Serra-Gomez, L. Lyons, V. Kratky, J. Alonso-Mora, and L. Ferranti, "Rule-based lloyd algorithm for multi-robot motion planning and control with safety and convergence guarantees," 2024, arXiv preprint arXiv:2310.19511.