



Formation Control of UAVs and Mobile Robots Using Self-organized Communication Topologies

Weixu Zhu¹, Michael Allwright¹, Mary Katherine Heinrich¹,
Sinan Oğuz¹, Anders Lyhne Christensen²,
and Marco Dorigo¹

¹ IRIDIA, Université Libre de Bruxelles, Brussels, Belgium
{weixu.zhu,michael.allwright,mary.katherine.heinrich,sinan.oguz,
mdorigo}@ulb.ac.be

² SDU Biorobotics, The Mærsk Mc-Kinney Møller Institute,
University of Southern Denmark, Odense, Denmark
andc@mimi.sdu.dk

Abstract. Formation control in a robot swarm targets the overall swarm shape and relative positions of individual robots during navigation. Existing approaches often use a global reference or have limited topology flexibility. We propose a novel approach without these constraints, by extending the concept of ‘mergeable nervous systems’ to establish distributed asymmetric control via a self-organized wireless communication network. In simulated experiments with UAVs and mobile robots, we present a proof-of-concept for three sub-tasks of formation control: formation establishment, maintenance during motion, and deformation. We also assess the fault tolerance and scalability of our approach.

1 Introduction

We target the control of mobile multi-robot formations—in other words, the maintenance of a possibly adaptive shape during navigation, including both shape outline and relative positions of individuals. Formation control is more frequently studied in control theory than swarm robotics (cf. distinction pointed out by [20]). In swarm robotics, physical coordination with non-physical connections has been studied in flocking (e.g., [7]), where an amorphous group forms during motion, and in self-assembly without physical connections, which has been demonstrated for immobile shapes that are definite and static [18] or amorphous and adaptive [19]. In these approaches, flexibility of individual robot positions has been used as a feature, similar to formation-containment control (e.g., [6]) in control theory, which maintains an overall convex hull.

Formation control—maintaining both overall shape and individual relative positions—merits further study in swarm robotics. We propose an approach based on the existing ‘mergeable nervous systems’ (MNS) [12] concept.

The MNS concept combines aspects of centralized and decentralized control, via distributed asymmetric control over a communication graph formed exclusively by self-organization. Our method targets control of definite swarm shape and relative positions of individuals, in non-physically connected robots. Widely studied formation control approaches [1, 4, 11] primarily make use of formation-level central coordination, and include *leader-follower* [21] (including virtual leader [17]), *virtual structure* [9, 17], and *behavior-based* [2, 3]. Our proposed hybrid approach uses a virtual structure that is not only a reference coordinate frame and a target formation, but also a target topology of the communication network. Robots cede motion control to distributed leaders (i.e., parents) that are their immediate neighbors in the communication topology, rather than following a single shared leader. Similarly to *behavior-based* control, the target formation is not necessarily rigid, as the parents can adapt the motion control of their immediate followers (i.e., children) on the fly, during tasks such as obstacle avoidance.

We select the review by [11] to define the aims of our proof-of-concept experiments. Then, a comprehensive approach to formation control should include the following sub-tasks: 1) formation establishment from random positions, 2) formation maintenance during motion, and 3) formation ‘deformability’ [11] (i.e., updating the target formation on the fly) during obstacle avoidance. We test formation establishment (Sect. 3.1) with various shapes and sizes of target formations. For formation maintenance, we test time-and-position cooperative and reactive motion, in response to an external stimulus (Sect. 3.2). For formation deformability, we test a scenario requiring multiple updates to the target formation during obstacle-exposed navigation (Sect. 3.2). We also target beneficial features typically seen in self-organization. First, we test fault tolerance, in terms of formation recovery after robot failure (Sect. 3.3). Second, we test scalability (Sect. 3.4), in terms of convergence time during formation establishment (Sect. 3.1) and reaction time in response to an external stimulus (Sect. 3.2).

2 Methods

Our formation control approach is based on the ‘mergeable nervous systems’ (MNS) concept [12], previously demonstrated with physical connections among ground robots. Here, we extend the concept to non-physical connections, with self-organized wireless communication topologies in a heterogeneous swarm.

Target Topology, Target Formation, and Motion Control. In our approach, an MNS is a set of robots connected in a self-organized wireless communication network, specifically a directed rooted tree, where the root acts as the brain robot of the MNS. A self-organization process results in a network with a given target topology. This network is used to execute distributed motion control, to move robots to positions and orientations that match a given target formation. The target topology is represented in graph G , and target formation is represented by a set of attributes A associated to the links of G . For each link between a parent robot and child robot, A includes the child target position

and orientation, relative to the parent, and includes the robot type of the child (either UAV or ground robot). A robot uses the full G and A as reference only if it is currently a brain. G and A are defined externally and can be updated during runtime. A non-brain robot receives a portion of the target from its parent, to use as its new reference. Specifically, robot r_n receives G'_n , the subgraph downstream from it, and the associated subset A_n .

To establish the target formation and maintain it during motion, each child cedes motion control to its parent, which directs it to the relative position and orientation indicated in A . The motion instructions communicate linear and angular velocities, with the parent as reference frame, via the following: 1) new linear velocity vector \mathbf{v} , magnitude in m/s; 2) new angular velocity vector $\boldsymbol{\omega}$, magnitude in rad/s; and 3) current orientation in unit quaternion \mathbf{q}_t , representing rotation axis and angle. To execute the instructions, the child first rotates \mathbf{v} and $\boldsymbol{\omega}$ by $-\mathbf{q}_t$, resulting in new vectors \mathbf{v}_q and $\boldsymbol{\omega}_q$, then begins moving in direction \mathbf{v}_q at speed $\|\mathbf{v}_q\|$ m/s while rotating around $\boldsymbol{\omega}_q$ at speed $\|\boldsymbol{\omega}_q\|$ rad/s. In order to calculate instructions that will move the child towards the target, the parent senses its child's current displacement vector \mathbf{d}_t and orientation \mathbf{q}_t , with itself as reference frame. At each step, the parent sends new motion instructions after calculating a new desired displacement \mathbf{d}_{t+1} and orientation \mathbf{q}_{t+1} for the child, and then calculating \mathbf{v} and $\boldsymbol{\omega}$ according to Eq. 1, as follows:

$$\mathbf{v} = k_1 \left(\frac{\mathbf{d}_{t+1} - \mathbf{d}_t}{\|\mathbf{d}_{t+1} - \mathbf{d}_t\|} \right), \quad \boldsymbol{\omega} = k_2 \cdot \|f(\mathbf{q}_{t+1}^{-1} \times \mathbf{q}_t)\|, \quad (1)$$

where k_1 and k_2 are speed constants, and where function $f(x)$ converts a quaternion to an Euler angle.

Formation Establishment and Maintenance. A target topology is established by robots forming directed communication links, becoming members of the same MNS. MNS topologies are self-organized via distributed *recruitment* operations and *handover* operations. Recruitment operations form new links. A robot tries to form new links with another robots if these two robots are not in the same network. Handover operations redistribute robots if their current topology nodes do not match the target G and A . A robot may handover its children to its parent or other children based on its G and A to change the topology of the network. Regardless of how robots are initially recruited, those at incorrect nodes will be shifted along the topology until all robots match the target G and A . In case of faulty robots, recruitment and handover operations also restore the target topology. When a robot is moving, it sends motion instruction to its children to maintain their relative positions and orientations according to A . A robot reacting to an external signal may send emergency motion instructions to its neighbors, and the instructions propagate through the MNS. The MNS is ‘deformable’ [11] (i.e., can switch the target formation on the fly) by updating the target G and A in the brain.

2.1 Experiment Setup

We run experiments with the multi-robot simulator ARGoS [16], using kinematic robot control. The arena is $10 \times 10 \times 2.5 \text{ m}^3$, fully enclosed, and optionally includes $0.04 \times 0.04 \times 0.02 \text{ m}^3$ static obstacles. The UAV model is based on the DJI F540 multi-rotor frame, which we extend with four ground-facing cameras. We limit UAV speed to 0.1 m/s, to match the 0.1 m/s maximum speed of the ground robots. UAVs maintain a 1.5 m altitude, after taking off at the start of an experiment. The ground robot model is an extended e-puck [8, 13, 14], with a fiducial marker ($0.03 \times 0.03 \text{ m}^2$ AprilTag [15]) encoding the robot ID. Obstacles also have AprilTags, encoding an obstacle identifier. At 1.5 m altitude, a UAV reliably views a ground area of $1.5 \times 1.5 \text{ m}^2$, detecting positions and orientations of ground robots and obstacles. If two UAVs are connected, they detect each other via ‘virtual sensing’ [10]—they each infer the other’s position and orientation relative to a mutually detected ground robot. Our setup assumes restriction to short-range communication. Messages can only be passed between robots if they are connected in the graph G , or if one is in the other’s field of view.

We run experiments (video available¹) for formation establishment, obstacle avoidance, fault tolerance, and scalability. We define and use nine target formations (F1–F9, see Fig. 1(a)). We conduct 100 runs per experiment and record robot positions throughout. We assess performance via ‘position error’ [11]—i.e., the difference between actual relative positions and those indicated in the target formation.

3 Results

3.1 Formation Establishment

We first test formation establishment from random starting positions. Establishment is considered successful if all robots merge into a single MNS and the robots achieve the given target formation. We test all nine target formations (100 runs each). Using the robots’ final node allocations after experiment termination, we use Euclidean distance to calculate position error E at each timestep, as follows:

$$E = \frac{1}{n} \sum_{i=1}^n E_i, \quad E_i = |d(\mathbf{p}_i - \mathbf{p}_1) - d(\mathbf{r}_i - \mathbf{r}_1)|, \quad (2)$$

where n is the total number of robots, \mathbf{p}_i is a robot’s current position, \mathbf{r}_i is a robot’s target position, and $i = 1$ is the brain. E_i for the brain is always zero, because the brain’s relative position to itself is constant. Position error E over time is given in Fig. 1(d), for all nine target formations. In all runs, the swarm successfully establishes the target formation within 400 s. The larger the swarm size, the more time it takes to converge. On average, convergence time is 12.79 s per robot (standard deviation of 5.32 s).

¹ <http://iridia.ulb.ac.be/supp/IridiaSupp2020-006/index.html>.

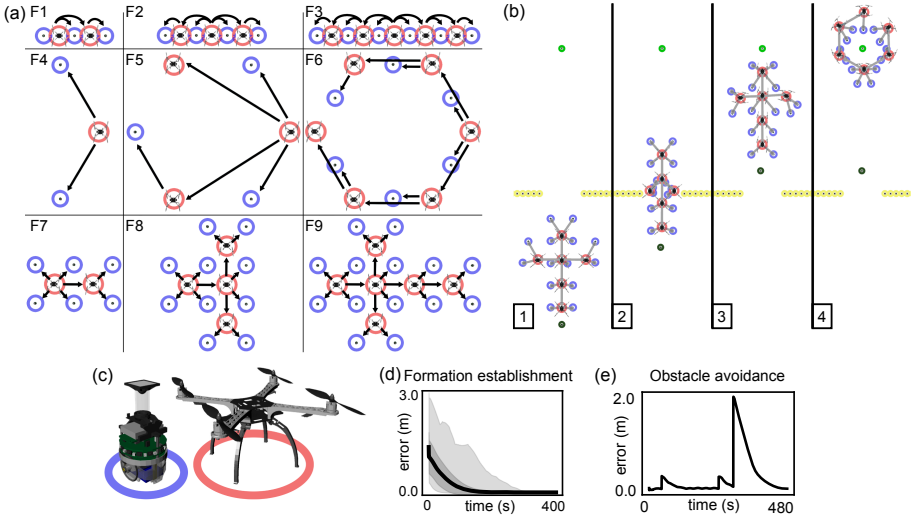


Fig. 1. (a) Target formations (F1–F9). Red circles are UAVs, blue are ground robots. (b) Formation-level obstacle avoidance scenario (screenshot from simulator); deformation. (c) Ground robot and UAV. (d) Formation establishment results. Average position error E over time for all target formations (900 runs total). Dark grey shows standard deviation; light grey shows maximum and minimum. (e) Formation-level obstacle avoidance results from example run. Position error E over time.

3.2 Formation-Level Obstacle Avoidance

We test ‘deformability’ [11]—i.e., whether the target formation can be updated on the fly, for instance by switching from a cross-shaped formation to a circular formation. For deformation, the brain updates the target topology and formation. Deformation is successful if the MNS establishes the new target formation after an update, such that the position error E returns to its prior level (approximately 0.1 m position error). In this experiment type, we define a wall with a narrow opening (a complex obstacle for formation control [11]) and a small box to be encircled. We use a shepherd robot as stimulus. In step 1, see Fig. 1(b), the MNS is in formation F9 and moves towards the wall because of the shepherd robot. In step 2, it switches from the cross-shaped formation F9 to a more elongated formation similar to formation F3, passing the opening. In step 3, it switches back to formation F9. In step 4, it encounters the small box, and switches to a circular formation similar to formation F6, surrounding the box. Position error E (see Eq. 2) over time is given in Fig. 1(e). Peaks occur when the target formation switches; the largest peak corresponds to the largest difference between the old and new formations. In all 100 runs, E returns to its prior stable level, after each formation switch.

3.3 Fault Tolerance

We test recovery of the topology and formation when a robot fails—i.e., its communication links break and it is arbitrarily displaced. Recovery is successful if position error returns to its prior level, from before the failure. Searching for robots is not within the scope of this paper, so the failed robot is displaced to a random position within the MNS’s field of view. Our approach is tolerant even to brain failure, as any robot can be replaced by its topologically closest neighbor. With 100 runs each, we test failure of a leaf node (Fig. 2(a,d)); a non-brain inner node (Fig. 2(b,e)); and a brain (Fig. 2(c,f)). We begin the assessment of each experiment at timestep 280 s, once all robots have established formation F9. In this target formation, all leaf nodes are ground robots and all inner nodes are UAVs. From the set of robots that are candidates for failure in the respective experiment (e.g., those at leaf nodes), one robot is randomly selected as the failed robot, and is removed and displaced at timestep 300 s. Shaded plots of position error E (see Eq. 2) over time are given in Fig. 2(a–c), and scatter plots of recovery time in relation to displacement distance of the failed robot are given in Fig. 2(d–f). Results show that the closer the failed robot is to the brain topologically, the longer the time to recover (note the different scale on the y-axes of Fig. 2(a–c)), and the less direct the relationship between displacement distance and recovery time. This weaker relationship reflects the increased difficulty of recovery, when the failure is closer to the brain. The MNS succeeds in fully recovering in 99% of 300 runs. In the remaining 1%, one ground robot erroneously moves slightly out of view; as searching for robots is not part of the experiment setup, it remains out of view.

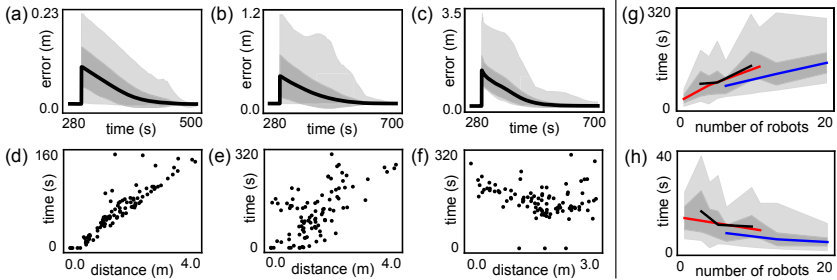


Fig. 2. (a–f) Formation recovery after three failure types: (a,d) leaf node, (b,e) non-brain inner node, and (c,f) brain. (a–c) Position error E over time, for each failure type (100 runs each). (d–f) Relationship between recovery time and displacement distance, for each failure type (100 runs each). (g–i) Scalability analysis. (g) Convergence time by number of robots. Each color line indicates average time for a shape type (shape types F1–3, F4–6, and F7–9 in Fig. 1(a)). Dark grey is standard deviation for all shape types; light grey, maximum and minimum. (h) Convergence time per robot, by number of robots. (Color indications match (g).)

3.4 Scalability

We assess scalability in terms of the initial time to converge on the target formation (in Sect. 3.1), and the reaction time during motion while the formation is being maintained (in Sect. 3.2). The total time to converge (see Fig. 2(g)) tends to increase sublinearly with increasing number of robots—in other words, the system scales slightly better than linearly. Convergence time per robot (see Fig. 2(h)) tends to decrease as the number of robots increases. These tendencies occur because merges often happen in parallel early in the establishment process. The formation shape also impacts convergence time; as this is a multidimensional variable, a comprehensive understanding would require further study.

In a physical MNS, reaction time depends on the number of robots a message passes through [12]. For our wireless MNS, we find that reaction time increases linearly according to the number of links from the stimulated robot to the furthest robot. Currently, there is no spread; one message takes one simulation step (200 ms). In real robots, message time will likely vary. In the experiments of [12], the real message rate was 100 ms (half the rate we set in simulation), using messages of comparable size. For wireless communication, a candidate for our setup would be Zigbee, with effective bit rate of 250 kbps [5].

4 Discussion and Conclusions

We have proposed a self-organized approach to formation control based on the existing concept of ‘mergeable nervous systems,’ which combines aspects of centralized and decentralized control. Robots in a swarm execute distributed asymmetric control via self-organized communication topologies. In simulated experiments we have demonstrated a successful proof-of-concept, showing that our approach can enable a swarm to establish and maintain a given formation while avoiding obstacles. We have demonstrated that, using the self-organized communication topology, the formation can recover after robot failure and displacement, and also can switch to a new formation on the fly. Although these are promising results, more comprehensive study is required to define the limits of these features, give formal guarantees, and systematically compare the performance of our method to other formation control approaches. In order to move our approach to real-robot experiments, future developments will need to address conditions such as sensor noise and communication latency, and add a layer of dynamic control in addition to kinematic control for the UAVs. Overall, we draw the conclusion that in the tested experimental setup our MNS-based approach is capable of fault-tolerant and scalable formation control during navigation, in a heterogeneous robot swarm comprising UAVs and ground robots.

Acknowledgements. This work is partially supported by the Program of Concerted Research Actions (ARC) of the Université libre de Bruxelles, by the Office of Naval Research Global (Award N62909-19-1-2024), by the European Union’s Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 846009, and by the China Scholarship Council Award No 201706270186.

Marco Dorigo and Mary Katherine Heinrich acknowledge support from the Belgian F.R.S.-FNRS, of which they are a Research Director and a Postdoctoral Researcher respectively.

References

1. Anderson, B.D., Fidan, B., Yu, C., Walle, D.: UAV formation control: theory and application. In: Blondel, V.D., Boyd, S.P., Kimura, H. (eds.) *Recent Advances in Learning and Control. Lecture Notes in Control and Information Sciences*, vol. 371, pp. 15–33. Springer, London (2008). https://doi.org/10.1007/978-1-84800-155-8_2
2. Balch, T., Arkin, R.C.: Behavior-based formation control for multirobot teams. *IEEE Trans. Robot. Autom.* **14**(6), 926–939 (1998)
3. Cao, Z., Xie, L., Zhang, B., Wang, S., Tan, M.: Formation constrained multi-robot system in unknown environments. In: *IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, vol. 1, pp. 735–740. IEEE (2003)
4. Chen, Y.Q., Wang, Z.: Formation control: a review and a new consideration. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3181–3186. IEEE (2005)
5. Cox, D., Jovanov, E., Milenkovic, A.: Time synchronization for ZigBee networks. In: *Proceedings of the Thirty-Seventh Southeastern Symposium on System Theory, SSST 2005*, pp. 135–138. IEEE (2005)
6. Dong, X., Hua, Y., Zhou, Y., Ren, Z., Zhong, Y.: Theory and experiment on formation-containment control of multiple multirotor unmanned aerial vehicle systems. *IEEE Trans. Autom. Sci. Eng.* **16**(1), 229–240 (2018)
7. Ferrante, E., Turgut, A.E., Huepe, C., Stranieri, A., Pinciroli, C., Dorigo, M.: Self-organized flocking with a mobile robot swarm: a novel motion control method. *Adapt. Behav.* **20**(6), 460–477 (2012)
8. Gutiérrez, Á., Campo, A., Dorigo, M., Donate, J., Monasterio-Huelin, F., Magdalena, L.: Open e-puck range & bearing miniaturized board for local communication in swarm robotics. In: *IEEE International Conference on Robotics and Automation*, pp. 3111–3116. IEEE (2009)
9. Lewis, M.A., Tan, K.H.: High precision formation control of mobile robots using virtual structures. *Autonom. Robots* **4**(4), 387–403 (1997). <https://doi.org/10.1023/A:1008814708459>
10. Liu, L., Kuo, S.M., Zhou, M.: Virtual sensing techniques and their applications. In: *International Conference on Networking, Sensing and Control*, pp. 31–36. IEEE (2009)
11. Liu, Y., Bucknall, R.: A survey of formation control and motion planning of multiple unmanned vehicles. *Robotica* **36**(7), 1019–1047 (2018)
12. Mathews, N., Christensen, A.L., O’Grady, R., Mondada, F., Dorigo, M.: Mergeable nervous systems for robots. *Nature Commun.* **8**, 439 (2017)
13. Millard, A.G., et al.: The Pi-puck extension board: a Raspberry Pi interface for the e-puck robot platform. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 741–748. IEEE (2017)
14. Mondada, F., Bonani, et al.: The e-puck, a robot designed for education in engineering. In: *Proceedings of the 9th Conference on Autonomous Robot Systems and Competitions*, vol. 1, pp. 59–65. IPCB: Instituto Politécnico de Castelo Branco (2009)

15. Olson, E.: AprilTag: a robust and flexible visual fiducial system. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), pp. 3400–3407. IEEE, May 2011
16. Pinciroli, C., et al.: ARGoS: a modular, parallel, multi-engine simulator for multi-robot systems. *Swarm Intell.* **6**(4), 271–295 (2012). <https://doi.org/10.1007/s11721-012-0072-5>
17. Ren, W., Sorensen, N.: Distributed coordination architecture for multi-robot formation control. *Robot. Auton. Syst.* **56**(4), 324–333 (2008)
18. Rubenstein, M., Cornejo, A., Nagpal, R.: Programmable self-assembly in a thousand-robot swarm. *Science* **345**(6198), 795–799 (2014)
19. Soorati, M.D., Heinrich, M.K., Ghofrani, J., Zahadat, P., Hamann, H.: Photomorphogenesis for robot self-assembly: adaptivity, collective decision-making, and self-repair. *Bioinspir. Biomim.* **14**(5), 056006 (2019)
20. Valentini, G., Ferrante, E., Dorigo, M.: The best-of-n problem in robot swarms: formalization, state of the art, and novel perspectives. *Front. Robot. AI* **4**, 9 (2017)
21. Wang, P.K.: Navigation strategies for multiple autonomous mobile robots moving in formation. *J. Robot. Syst.* **8**(2), 177–195 (1991)