

# Highly Scalable Frontier-Led Chain Formations for Search and Exploration Problems

Danny Roberts, Simos Gerasimou, and Alan Millard

Department of Computer Science, University of York, York, UK

`danny.roberts@york.ac.uk`

`simos.gerasimou@york.ac.uk`

`alan.millard@york.ac.uk`

**Abstract.** Building upon the frontier-led swarming concept introduced by Tran[6], this paper presents a novel adaptation that integrates a rigid chain formation with traditional flocking for the exploration and coverage of unknown environments. This approach augments Tran’s method by guiding flocking dynamics into more structured outcomes through a rigid chain formation, maintaining the system’s robustness to agent failure while achieving near-linear scalability. The agents in this system identify and maintain explicit links with their immediate neighbours, allowing for a more organized exploration strategy that complements the frontier-led approach. [Summarize results here](#). Future work will explore the capability of the chain to intelligently navigate around obstacles by splitting and reuniting, further solidifying this method’s potential in applications such as humanitarian demining and ocean garbage clean-up. This research aims to offer a new perspective on the integration of rigid formations within swarm robotics, proposing a balanced approach between structured organization and the adaptive and scalable qualities of swarms.

**Keywords:** Swarm Robotics · Scalability · Formations

## 1 Introduction

Swarm intelligence, inspired by the collective behaviours of natural systems like colonies of ants and flocks of birds [1], aims to capture the powerful emergent behaviours that come with a system made up of simple agents[4]. Leveraging the principles of decentralized control and local interaction, swarm robotic systems offer unparalleled scalability, robustness, and flexibility, making them ideal for complex tasks ranging from environmental monitoring to search and rescue operations [2,3]. Despite these advantages, challenges remain in maintaining effective communication and cohesive formations without prior knowledge of the terrain, a critical factor in the efficiency and success of swarm-based missions [7]. This research builds upon the foundation of “frontier-led swarming”, introduced by Tran et al. [6], to address these challenges through the novel integration of rigid chain formations, aiming to enhance both the structural integrity and scalability of swarm operations.

## 2 Related Work

### 2.1 Swarm Algorithms

### 2.2 Area coverage / search / exploration

## 3 Proposed Algorithm

### 3.1 Establishing the Basis

Consider a set of  $n$  agents  $A^1, A^2 \dots A^n$  where  $p^i$  and  $v^i$  represent the position and velocity of  $A^i$ . We define a neighbour as another agent within a distance  $d_x$  of the agent in question, with the set of all neighbours of  $A^i$  represented by  $N_x^i$ :

$$N_x^i = \{A^j \in A \setminus \{A^i\} : \|p^i - p^j\| \leq d_x\} \quad (1)$$

**Flocking Forces** At each time step  $t$  a force is exerted on an agent based on the current state of its nearby neighbours. To implement Reynolds Boids [5], we need the following forces per agent  $A^i$ . Each force has its own proximity radius  $d_i$ ; e.g.  $d_{coh}$  represents the cohesion radius.

- (2) **Cohesion** (pull towards the average position of its neighbours)
- (3) **Alignment** (pull towards matching the velocity of its neighbours)
- (4) **Separation** (push away from neighbours that are too close).

$$F_{coh}^i = \left( \frac{1}{|N_{coh}^i|} \sum_{j \in N_{coh}^i} p^j \right) - p^i \quad (2)$$

$$F_{ali}^i = \left( \frac{1}{|N_{ali}^i|} \sum_{j \in N_{ali}^i} v^j \right) - v^i \quad (3)$$

$$F_{sep}^i = \sum_{j \in N_{sep}^i} \frac{p^i - p^j}{\|p^i - p^j\|^3} \quad (4)$$

**Goal Forces** "Frontier-led Swarming" [6] combines basic flocking dynamics with a frontier generated from a local map. This gives agents a goal in addition to maintaining the flock. Tran's work demonstrates an algorithm for generating this frontier cell, which we build on in our extension of their algorithm. By subtracting an agent's current position from the frontier goal point  $p^{goal}$ , we can calculate a goal force  $F_{goal}$  for each agent.

$$F_{goal}^i = p^{goal} - p^i \quad (5)$$

### 3.2 Chain formation

The goal of a chain formation is to spread out across a frontier, such that the swarm as a whole can move in formation all covering new ground at once. In order to encourage spreading out in a line vs a cloud, we can apply a force to agents that are in front or behind of their peers. By using the dot product of an agent's velocity (or vector representing their heading) we get a value representing how far ahead or behind of an agent's heading its peer is. The average of these forces for all neighbours along with the other flocking forces will "flatten" a flock into a line or chain formation.

$$F_{chn}^i = \frac{v^i}{|N_{chn}^i|} \left( \sum_{j \in N_{chn}^i} \hat{p}^j \cdot \hat{v}^i \right) \quad (6)$$

### 3.3 Final Algorithm

When running the algorithm we combine the forces above, using two parameters for each force:  $d_x$  for the radius of influence of that force, and  $w_x$  for the weighting of that force. These weights and distances will need to be tuned for specific types of robot with different properties such as maximum speed and acceleration. Furthermore, depending on the current goal of the system, these weights and distances could be dynamically changed to close in on a target or spread out and explore a larger area. After the force vector has been applied to the velocity,

$$F_t^i = w_{coh} F_{coh}^i + w_{ali} F_{ali}^i + w_{sep} F_{sep}^i + w_{goal} F_{goal}^i + w_{chn} F_{chn}^i \quad (7)$$

$$v_{t+1}^i = v_t^i + F_t^i \quad (8) \quad p_{t+1}^i = p_t^i + v_t^i \quad (9)$$

*NB: Velocity is capped. If  $\|v^i\| > v_{cap}$  then the magnitude will be set to  $v_{cap}$*

## 4 Experimental Methodology

### 4.1 Algorithms for Comparison

### 4.2 Treatments

### 4.3 Data Collected

The main data collection target for these simulations will be **time to explore the environment**. This will be split into sub-measurements (80%, 95% and 99% coverage), as well as an overall exploration rate value in  $m^2/s$  collected throughout the simulation. These values will give a clear picture of how the system performs throughout its runtime rather than simply how long complete coverage takes. This is advantageous as complete coverage time can vary wildly if a small amount of space (in a corner for example) is missed on the first pass, and faster but slightly less complete coverage can be acceptable trade-off depending on the scenario. In addition, several other non-goal metrics will be collected:

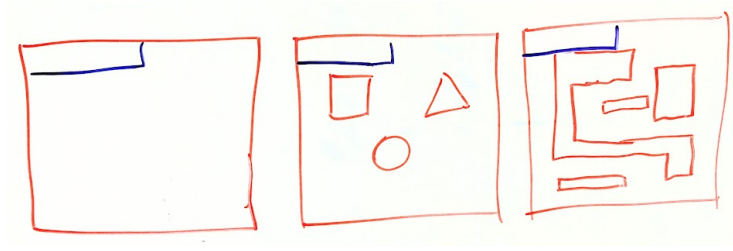


Fig. 1: Environments of varying complexity used for testing. Blue area indicates starting area of the swarm This image is obviously just a mock-up, I'll make a proper diagram later

*Messages passed and communication bandwidth* are crucial metrics when it comes to scalability of the system, as in many systems message passing bandwidth of a given agent will scale  $O(N^2)$  where  $N$  is the number of agents. A scalable system should have a message passing bandwidth of  $O(N)$  to allow for unlimited expansion.

*Message passing distances* directly determine the amount of radio power required to send a message, and crucially this also scales with  $O(D^2)$ . Therefore, keeping distance low has a significant impact on required transmit power.

*There's more...*

## 5 Results

Table 1: Time to complete exploration (99%)

Algorithm	Open Environment	Simple Obstacles	Complex Environment
My Algo	82s	...	...
Tran-like	91s	...	...
Naive algo	120s	...	...
Theoretical optima	40s	...	...

I'd describe the results here if I had any that weren't made up  
 Maybe I should just focus on the open or the simple environment for now,  
 so I can stick to analysing scalability?

## References

1. Garnier, S., Gautrais, J., Theraulaz, G.: The biological principles of swarm intelligence. *Swarm Intelligence* **1**(1), 3–31 (Jun 2007)
2. Hamann, H.: Superlinear Scalability in Parallel Computing and Multi-robot Systems: Shared Resources, Collaboration, and Network Topology. In: Berekovic, M., Buchty, R., Hamann, H., Koch, D., Pionteck, T. (eds.) *Architecture of Computing Systems – ARCS 2018*. pp. 31–42. Springer International Publishing, Cham (2018)
3. Ijspeert, A.J., Martinoli, A., Billard, A., Gambardella, L.M.: Collaboration Through the Exploitation of Local Interactions in Autonomous Collective Robotics: The Stick Pulling Experiment. *Autonomous Robots* **11**(2), 149–171 (Sep 2001)
4. Mataric, M.J.: From Local Interactions to Collective Intelligence. In: Steels, L. (ed.) *The Biology and Technology of Intelligent Autonomous Agents*. pp. 275–295. Springer, Berlin, Heidelberg (1995)
5. Reynolds, C.W.: Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH Computer Graphics* **21**(4), 25–34 (Aug 1987)
6. Tran, V.P., Garratt, M.A., Kasmarik, K., Anavatti, S.G., Abpekar, S.: Frontier-led swarming: Robust multi-robot coverage of unknown environments. *Swarm and Evolutionary Computation* **75**, 101171 (Dec 2022)
7. Xu, D., Zhang, X., Zhu, Z., Chen, C., Yang, P.: Behavior-Based Formation Control of Swarm Robots. *Mathematical Problems in Engineering* **2014**, e205759 (Jun 2014)