



Effect of swarm density on collective tracking performance

Hian Lee Kwa^{1,2} · Julien Philippot³ · Roland Bouffanais³

Received: 29 July 2022 / Accepted: 26 February 2023 / Published online: 21 March 2023

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract

How does the size of a swarm affect its collective action? Despite being arguably a key parameter, no systematic and satisfactory guiding principles exist to select the number of units required for a given task and environment. Even when limited by practical considerations, system designers should endeavor to identify what a reasonable swarm size should be. Here, we show that this fundamental question is closely linked to that of selecting an appropriate swarm density. Our analysis of the influence of density on the collective performance of a target tracking task reveals different ‘phases’ corresponding to markedly distinct group dynamics. We identify a ‘transition’ phase, in which a complex emergent collective response arises. Interestingly, the collective dynamics within this transition phase exhibit a clear trade-off between exploratory actions and exploitative ones. We show that at any density, the exploration–exploitation balance can be adjusted to maximize the system’s performance through various means, such as by changing the level of connectivity between agents. While the density is the primary factor to be considered, it should not be the sole one to be accounted for when sizing the system. Due to the inherent finite-size effects present in physical systems, we establish that the number of constituents primarily affects system-level properties such as exploitation in the transition phase. These results illustrate that instead of learning and optimizing a swarm’s behavior for a specific set of task parameters, further work should instead concentrate on learning to be adaptive, thereby endowing the swarm with the highly desirable feature of being able to operate effectively over a wide range of circumstances.

Keywords Adaptivity · Multi-agent systems · Swarm density · Swarm robotics · Target tracking

✉ Hian Lee Kwa
hianlee_kwa@mymail.sutd.edu.sg

✉ Roland Bouffanais
roland.bouffanais@uottawa.ca

¹ Engineering Product Development, Singapore University of Technology and Design, Singapore, Singapore

² Thales Research and Technology, Thales Solutions Asia, Singapore, Singapore

³ Department of Mechanical Engineering, University of Ottawa, Ottawa, ON, Canada

1 Introduction

Decentralized multi-robot systems (MRS) have recently become a highly active field of study owing to their unique ability to tackle a number of critical and challenging problems where a task has to be accomplished in a highly dynamic environment. When facing rapidly evolving conditions, the effectiveness of an MRS critically hinges on its ability to change behavior owing to its flexibility and/or adaptivity Dorigo et al. (2021). This includes a wide range of problems such as target tracking (Coquet et al., 2019, 2021), area protection (Strickland et al., 2018; Shishika & Paley, 2019), dynamic area monitoring (Vallegra et al., 2018; Zoss et al., 2018), area mapping (Kit et al., 2019; Crosscombe & Lawry, 2021; Liu et al., 2021), and environment classification (Ebert et al., 2018, 2020). Beyond flexibility and adaptivity, the effectiveness of an MRS has also been shown to be related to robustness and scalability (Bouffanais, 2016; Hamann, 2018b; Dorigo et al., 2021). These key attributes stem from the lack of a central controller that dictates the actions of the individual agents and have made MRS very attractive when considering operations in some of the most challenging circumstances involving expansive, unstructured, and dynamic environments. The decentralized nature of such systems divides up the overall control of the MRS and gives individual robots the ability to make decisions based on their local environment. This allows them to: (1) quickly react to changes in their surroundings (flexibility), (2) continue operations despite agent failures given the lack of a central point of control (robustness), and (3) eliminate potential processing bottlenecks (scalability). In addition, an MRS also has the ability to learn or change its behavior to deal with new circumstances (adaptivity).

One key design consideration and a critical question that practitioners must contend with is selecting the number of robotic units required to achieve a desired performance for a given task and environment (Hamann, 2018b). Indeed, it has been shown that in both biological and engineered multi-agent systems, effective emergent collective actions require a critical number of swarming agents (Khaluf et al., 2017; Schranz et al., 2021). Specifically, Schranz et al. (2021) stated that “the advantages of swarm intelligence algorithms can only be exploited if a critical mass of swarm members is reached.” However, a system’s collective behavior is not the only consideration when determining a swarm’s size. For a given task, several other aspects also influence the selection of the number of swarming agents, such as: (1) system scalability, (2) technical capabilities of the individual agents (e.g., communications range, sensor range, maneuverability, maximum speed, etc.), and (3) financial or logistical constraints (i.e., the number of robots that can be built, stored, and operated given the available resources) (Schroeder et al., 2019). It can therefore be said that selecting the number of robotic units is far from being a trivial task for system designers.

In the literature, MRS come in various shapes and sizes, with their sizes covering three orders of magnitude. These range from the 1024-unit Kilobot swarm by Rubenstein et al. (2014), to the 150 unit subCULTron system by Thenius et al. (2016), to the hunter drone swarm by de Souza et al. (2022) that can operate with less than 10 units. It should be noted that there is no consensus on what constitutes a robot swarm in terms of the number of units (Hamann, 2018b). However, in the vast majority of works, it is often unclear how exactly this quantity is selected. Would a half-sized Kilobot swarm perform vastly differently from the full-sized Kilobot MRS (Rausch et al., 2019)? At the other end of the spectrum, the same question can be posed to system designers who utilize relatively small systems, e.g., with 10 or less units. While it is often argued that such a small system size

would be sufficient for the task at hand (Zoss et al., 2018; Kit et al., 2019; Kwa et al., 2021), one is often left wondering if the limit in the actual number is not related to other factors, such as the logistical, financial, or technical challenges discussed above.

The size of the system is inextricably linked to the density of agents. Many works hint at the existence of a minimum critical density required to yield an effective self-organization of the swarm. For example, the agents used by the Kilobot swarm are only able to communicate with each other at a maximum distance of 10 cm (Rubenstein et al., 2014). Similarly, it was determined by Hornischer et al. (2020) that for a robot to function as intended in their system, a minimum of 3 other robots had to be located within an agent's communication range. Beyond the constraints in communications or information exchange, a scaling law has been used to quantify the effectiveness of the Bunch of Buoys (BoB) environmental monitoring system when performing a dynamic area monitoring task with 50 buoys for a pre-determined surface area (Zoss et al., 2018).

Fundamentally, the question of the selection of an MRS size—or equivalently its agents density—for a given task has never been systematically and satisfactorily answered, if at all. As already mentioned, the importance of a minimum density has long been acknowledged to ensure the emergence of a desired collective behavior in simplistic physics models. The self-propelled particles (SPP) in Vicsek's model is a paradigmatic example of that (Vicsek et al., 1995). Irrespective of the amount of noise present, it was shown that the agent density is key in ensuring the self-ordering process. Specifically, at low density, the swarm of SPP exhibits a disordered state that undergoes a phase transition leading to an ordered phase—with all SPP being practically aligned—when increasing the density of agents. Beyond the simplistic case of SPP seeking to align their direction of travel, Hamann (2012) considered the effect of swarm size and density—the surface area being kept constant—on system-level performance for several collective behaviors and decision-making protocols. This revealed superlinear performance increases when the system was enlarged, followed by a decrease in performance once the swarm exceeded a certain size. This degradation in performance was attributed to the increasing effect of interference on the system's performance, resulting from the increased amount of collision avoidance behavior carried out by the agents.

Besides collective motion, the problem of a system's exploration–exploitation balance, which is an important factor in determining the overall swarm performance, also needs to be considered. In various computational optimization tasks, such as virtual agents carrying out particle swarm optimization, and in the operation of MRS, larger swarms have been shown to exhibit higher levels of exploratory activity (Shishika & Paley, 2019; Piotrowski et al., 2020). However, this large amount of exploration needs to be paired with an adequate amount of exploitation to make use of the information gathered during the exploration process.

In this paper, we study the effects of density on the effectiveness in target tracking by an MRS: a highly dynamic collective task. Given the importance of a swarm's exploration and exploitation balance, we also examine the influence of swarm density on this balance, and show how it ultimately affects the overall system performance. It can be argued that in past studies, some of the tasks considered were too undemanding and could be accomplished using independently operating units, without the need for self-organization or emergent collective actions. However, we believe that it is only through dealing with such challenging problems that a system can demonstrate the full power of swarm intelligence. Therefore, this study is conducted using a fast-moving target (i.e., one that moves faster than any of the swarm's component agents) that does not emit a signal gradient field, making it akin to a visual search task where a target

needs to be seen to have its presence confirmed (Esterle & Lewis, 2020). Doing so eliminates the possibility of using simplistic methods such as gradient descent to track the target, thus forcing the system to self-organize and behave as a collective to track the target. It should be noted that while there are stochastic methods that attempt to control agent density, such as those using Markov processes (Açikmeşe & Bayard, 2012; Elamvazhuthi & Berman, 2019; Biswal et al., 2021), in this paper, we focus mainly on strategies employing deterministic approaches.

Our investigation reveals three density ‘phases’ that characterize the tracking ability of an MRS, namely: (1) a low density phase where the system is simply unable to coordinate the agents’ behavior to carry out the tracking task, (2) a transition phase where the performance of the system rises very quickly with increasing swarm density, and (3) a high density phase where the system is able to continuously track the target without any interruptions. These phases occur regardless of the number of agents within the system. We explain these trends by the existence of a system-level exploration–exploitation trade-off within the transition phase, with clearly identifiable exploration-limited and exploitation-limited regimes. Interestingly, the level of connectivity among agents can be used to skew the swarm dynamics towards either exploration or exploitation. Specifically, increasing (resp. decreasing) connectivity boosts the tracking performance in the low density exploitation-limited (resp. high density exploration-limited) regime. Nonetheless, the system size still plays a role in determining the ability of the MRS to carry out exploration and exploitation. Lastly, we discuss the significant ramifications that these results have for the development of effective collective dynamics through machine learning and automated design.

2 Related work

2.1 System size and exploration

The number of units employed within a multi-agent system is one of the basic parameters that a system designer needs to select. In MRS or virtual multi-agent systems (MAS), studies on a system’s size usually occur within an environment of a fixed size. As such, adding agents into a system equates is equivalent to an increase in agent density, and conversely removing agents results in a system with a lower swarm density. In the field of optimization, the number of agents, sometimes known as candidate solutions, employed is synonymous with the amount of exploration carried out by a system (Van Den Bergh & Engelbrecht, 2001; Piotrowski et al., 2020). As can easily be understood, using a larger number of agents allows the system to more thoroughly explore the search space, thereby reducing the chances of the optimization algorithm to settle in local optima. Similar observations were made when studying bee colonies in the wild where it was observed that larger groups were able to employ more scouts, thereby increasing the amount of exploration carried out and the accuracy of the colony’s final consensus (Schaerf et al., 2013). These results were also observed in engineered MRS, with system performances tending to increase with larger system sizes (Kit et al., 2019; Prasetyo et al., 2019; Shishika & Paley, 2019; Jurt et al., 2022). Similar to the virtual MAS used in optimization and those found in Nature, these

increases in performance are attributed to the increased exploration carried out by the larger systems.

2.2 Marginal utility

Given that larger MRS tend to yield systems with better performance, some studies have focused on the marginal utility gain when adding agents. These works focus on the marginal utility gain under the context of resource foraging. One of the earliest works on this topic was done by Lerman and Galstyan (2001), who demonstrated that while additional robots may indeed improve the foraging performance of the group as a whole, the amount of resources gathered by each robot, i.e., their individual efficiency, decreases. This is due to the increased level of physical interference in larger systems caused by the robots colliding with each other. Similar results were also demonstrated by Kit et al. (2019) in a collective area mapping task and by Sung et al. (2018, 2020) in a target tracking scenario. However, in target tracking, unlike resource foraging and area mapping, the swarm performance no longer increases after the MRS reaches a certain size. This is because at this point, the system is able to track the target regardless of how it moves. As such, the addition of units into the MRS provides no extra performance gain for the system. Interestingly, even in MAS optimization, it is often reported that increasing the number of agents beyond a certain point yields only marginal benefits to the system's accuracy and notably lengthens computational times (Roeva et al., 2015).

The reducing robot efficiency with increased system size and higher agent densities is known in economics circles as the “Law of Diminishing Marginal Returns” and was modeled in MAS by Hamann (2012). This was done using an exponential function that accounted for both the benefits of inter-agent cooperation and the detriments caused by the additional interference. Subsequently, Schroeder et al. (2019) combined this multi-agent performance strategy with a simple cost model to maximize the performance of their foraging MRS while minimizing the overall system cost by finding the ideal number of robots to be constructed for their system.

The diminished marginal utility caused by the addition of more agents may be further aggravated if common resources are to be shared and exploited. In a foraging task, Hecker and Moses (2015) concluded that when agents had to deposit their resources at a central location, individual robot efficiency decreased further in large swarms due to robots being required to travel longer distances to obtain resources. In Rosenfeld et al. (2006), the authors showed that the over-exploitation of common resources results in a higher level of physical interference between agents. In turn, this increased interference may result in negative marginal utility, i.e., the performance of the system decreases when more agents are included within the MRS. This occurs despite the higher amount of exploration taking place due to the increased swarm density and is because the advantages provided by the additional exploration is overshadowed by the exponentially larger level of interference encountered by the individual agents. This result was also demonstrated by Hamann (2018a) in a stick-pulling task, allowing the author to demonstrate three separate phases when adjusting a swarm's size: (1) a phase where agents do not collaborate enough and do not make full use of the available resources, (2) a phase where agents share too much information, resulting in the depletion of shared resources, and (3) an intermediate phase where the resources are neither underutilized nor overutilized, resulting in the optimal performance of the system.

2.3 Motility-induced phase separation

The increased amount of physical interference between agents is more apparent in systems tasked with clustering around certain points (e.g., target tracking and consensus tasks). As previously mentioned, high levels of clustering results in higher levels of physical interference in areas of high local swarm density. This in turn results in a positive feedback loop, causing more agents to aggregate in preexisting areas of high local agent densities in a phenomenon known as Motility-Induced Phase Separation (MIPS) (Cates & Tailleur, 2015). The buildup of agents in specific areas results in excessive amounts of exploitation, thus degrading the system's performance. To combat such excessive levels of exploitation caused by MIPS, several strategies have been developed to take into account the local swarm density around individual agents. When using an MRS to identify areas of high light intensity, modifications were made to the original BEECLUST algorithm, allowing robots to move off from its waiting position should it determine that it is located within an area with high agent density. This subsequently prevented excessive flocking of agents and encouraged a higher level of exploration, thereby making the system more responsive to dynamic environment changes (Wahby et al., 2019). Li et al. (2017) developed a strategy to control the positional distribution of robots in an MRS based on local swarm density and the strength of a signal field. To carry out a foraging task, Pang et al. (2019) varied the step size taken by each robot based on its local swarm density estimation. In target search and track tasks, many strategies have incorporated inter-agent repulsion to prevent the excessive flocking of robots at a single location, thereby promoting area exploration (Zhang et al., 2019; Dadgar et al., 2020; Kwa et al., 2020a).

3 Strategy and simulation

3.1 Search and track strategy

The strategy used to facilitate cooperative tracking of fast-moving targets was previously presented in (Kwa et al., 2021, 2022). This strategy consists of two main components: (1) agent aggregation augmented with a short-term memory and, (2) inter-agent adaptive repulsion, from which the system's adaptivity is derived. These two components are responsible for directing system exploitation of information and environment exploration, respectively, and is done by generating two velocity vectors at each time-step that are combined to give a final agent velocity vector:

$$\mathbf{v}_i[t] = \mathbf{v}_{i,\text{att}}[t] + \mathbf{v}_{i,\text{rep}}[t], \quad (1)$$

where $\mathbf{v}_{i,\text{att}}[t]$ and $\mathbf{v}_{i,\text{rep}}[t]$ are the velocity vectors generated by the attractive component and the repulsion component, respectively. The final velocity vector, $\mathbf{v}_i[t]$, is then scaled by v_{max} , the maximum speed of the agent. The overall exploration and exploitation balance, or exploration–exploitation dynamics (EED), of the system is controlled via the adjustment of the degree of the interconnecting topological k -nearest neighbor communications network. This adjustment and its effect on the EED will be further discussed in the next section.

The overall strategy employed in the system is summarized in Algorithm 1. For further details regarding the strategy's components, the reader is referred to Kwa et al. (2021) and Appendix 1.

Algorithm 1 : Dynamic k -Nearest Network Search and Tracking Strategy

```

1: Set  $t = 0$ ,  $k \in [2, N - 1]$ ,  $\omega = 1$ , and  $c = 0.5$ 
2: while System active do
3:   for All agents do
4:     Set point of attraction
5:     Calculate  $\mathbf{v}_{i,\text{att}}[t]$ 
6:     Calculate  $\mathbf{v}_{i,\text{rep}}[t]$ 
7:      $\mathbf{v}_i[t] \leftarrow \mathbf{v}_{\text{att},i}[t] + \mathbf{v}_{\text{rep},i}[t]$  // Apply Eq. (1)
8:      $\mathbf{v}_i[t] \leftarrow (v_{\text{max}}/v_i[t]) \cdot \mathbf{v}_i[t]$  // Ensure magnitude of velocity vector
        does not exceed maximum speed
9:      $\mathbf{x}_i[t + 1] \leftarrow \mathbf{x}_i[t] + \mathbf{v}_i[t]$  // Update agent position
10:   end for
11:    $t \leftarrow t + 1$ 
12: end while

```

3.2 Network connectivity

A key ingredient to any MRS is the communications network used by individual robots used to share information around the system (Sekunda et al., 2016). While most research on MRS have either assumed agents with unlimited communications range (Coquet et al., 2021; Rossides et al., 2021), or included all other robots within communications range (Dadgar et al., 2017; Jensen et al., 2018; Dadgar et al., 2020), it has been shown that tuning the network connectivity of a system can have considerable effects on a swarm's collective dynamics. Mateo et al. (2017) demonstrated that increased levels of communication resulted in reduced collective response to a simulated predator attack. Furthermore, Mateo et al. (2019); Horsevad et al. (2022b) also showed that while systems with higher levels of communication were more adept at collective mimicking of a slow-moving leader, lower levels of connectivity were required to more accurately replicate the movements of a fast-moving driving agent. Similar increases in performance were also observed in a scenario where agents were required to accurately characterize a static and noisy environment (Crosscombe & Lawry, 2021; Liu et al., 2021; Kwa & Bouffanais, 2022).

Building off these conclusions, we previously showed that there exists an optimal level of connectivity, k^* , of a k -nearest neighbor network at which the performance of a swarm is maximized while tracking a fast-moving target (Kwa et al., 2020a, 2022b, 2021, 2022). At this optimum, an MRS is able to attain the ideal balance between the level of exploratory and exploitative activities that its component agents carry out. This value of k^* changes depending on the conditions the MRS is operating in. It should be noted that a neighborhood is understood in the network sense; an agent i has as many neighbors as its degree k . Given that time-varying network topologies are considered, the neighborhoods of each individual agent are dynamic and change over time.

3.3 Problem statement

In this work, a set of N tracking agents $A = \{a_1, a_2, \dots, a_N\}$ and a single target, o , move within a bounded two-dimensional square search-space of dimensions $L \times L$ devoid of any obstacles, where $L \in [10^{0.6}, 10^{2.65}]$. The agents' and target's positions are denoted by $\mathbf{x}_i = (x_i, y_i)$, and maximum velocities of $\mathbf{v}_{a,\max}$ and $\mathbf{v}_{o,\max}$, respectively. It should be emphasized that in this work, we consider a target that moves faster than its pursuing agents. As such, $\mathbf{v}_{a,\max} < \mathbf{v}_{o,\max}$. In this work, the maximum velocities are set as follows: $\mathbf{v}_{a,\max} = 0.1$ and $\mathbf{v}_{o,\max} = 0.15$.

The target is modeled using a disc-shaped binary objective function with a fixed radius of $r = 1$. A target is considered to be tracked if an agent lies within its radius. Formally:

$$\text{cov}(o, t) = \begin{cases} 1, & \exists i \in A \text{ s.t. } \|\mathbf{x}_i - \mathbf{x}_o\| \leq r, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

The modeling of the target as a binary objective function, as done here, makes the problem more challenging and more similar to a visual search task where a target needs to be seen to have its presence confirmed (Hönig & Ayanian, 2016; Esterle & Lewis, 2020). This is in contrast to tracking a target from the intensity of an emitted signal (e.g., radio signal strength, chemical plume, etc.) in which various gradient-descent methods can be used. While such gradient-descent methods form one of the most widely used classes of techniques tracking targets, such techniques become completely ineffective when dealing with such binary objective functions. The use of this type of function ensures that such simplistic techniques cannot be used by the system. This conservative approach represents one of the most challenging cases with a near-zero-range sensor tracking a target that is moving faster than the agent themselves. As already mentioned, it can be argued that the full power of swarm intelligence can only be accessed when dealing with such challenging scenarios.

The target is set to move according a non-evasive movement policy. Using this policy, a target travels toward a randomly generated waypoint within the search space. Upon reaching this waypoint, another waypoint is generated, causing the target to change its direction of travel. This process is repeated until the end of the simulation.

The goal of the system is to maximize its tracking performance within the environment, given by the reward function:

$$\Xi = \frac{1}{T} \sum_{t=1}^T \text{cov}(o, t), \quad (3)$$

where T is the total time period of interest, set as $T = 100,000$. This large value of T was set to ensure the statistical stationarity of the results, i.e., the system that we are testing is ergodic and given that the conditions stay the same, its tracking performance will tend to a constant value when $T \rightarrow \infty$ (Khaluf et al., 2013). In the simulations, the agents are tasked with tracking the targets in an environment free from obstacles and are assumed to have perfect information about the target's location once within the target's radius. While carrying out the tracking task, the agents are also assumed to have perfect information about their pose in the environment with respect to a global reference frame.

3.4 Metrics

In addition to the MRS tracking performance, its collective dynamics can also be studied through the quantification of the EED of the system. As such, the reward function presented in Eq. (3) is used to quantify how well a system exploits the targets' positional information.

To further quantify a system's collective dynamics, its *exploration ratio* is also calculated. An agent is considered to be exploring the environment if it has entered the 'exploratory' state and $S_{i,\text{exp}}[t] = 1$. In this exploratory state, the agent tries to repel itself away from its neighbors, causing it to explore the environment in the process. In contrast, when an agent detects a target, it enters a 'tracking' state, $S_{i,\text{exp}}[t]$ is set to 0, and the agent attempts to follow the target. Therefore, the overall exploration ratio of the swarm is calculated as follows:

$$\Theta = \frac{1}{NT} \sum_{t=1}^T \sum_{i=1}^N S_{i,\text{exp}}[t], \quad (4)$$

where N is the total number of agents within the system. With the exploration ratio, a higher value of Θ indicates a higher proportion of agents spending more time searching for a target, and hence higher levels of exploration. Conversely, at lower Θ , agents attempt to move towards a target, thus spend more time exploiting target information.

The concept of system's *swarm density* or *agent density* is also introduced. It should be noted that these two terms are used interchangeably in this paper. This is calculated as follows:

$$\rho = N/L^2. \quad (5)$$

However, the global swarm density should not be taken at face value; it is very unlikely that the agents are evenly distributed across the environment. Instead, due to the flocking and dispersion of agents, the actual distribution of the swarm is likely to be nonuniform (Khaluf et al., 2017). As such, the concept of *local swarm density* is quantified by means of the measure of the average distance from each agent to its 6 nearest neighbors (see Appendix 2 for further details about the choice of this particular value). This allows for the calculation of the local swarm density for each agent. The local density is then averaged for all agents in the swarm across the entire duration of the simulation:

$$\rho_L = \frac{1}{NT} \sum_{t=1}^T \sum_{i=1}^N \frac{7}{\pi L_{i,t}^2}, \quad (6)$$

where L_i is the average distance from an agent to its six closest neighbors. Using ρ_L , one can quantify the level of clustering carried out by a system's agents. This is done by finding the difference between the swarm density and the average local swarm density: $\Delta\rho = \rho_L - \rho$, referred to as the density difference.

4 Results

4.1 Effect of swarm density on tracking performance

To investigate the effects of swarm density on tracking performance, we first run simulations of different swarm sizes, ranging from $N = 20$ to $N = 50$, tracking a non-evasive target in different environment sizes. In Fig. 1, three separate 'phases' can be

observed. At low density ($\rho \lesssim 10^{-3}$), the system is effectively unable to track the target. This is due to the large distances between agents, preventing the units from effectively clustering together and tracking the target. The large inter-agent distances also reduce the chances of an agent encountering the target, further aggravating the problem of poor tracking performances. It is worth stressing that a key assumption of our model is that the quality of information exchanges remain unaffected by the large distances between agents—the agents keep the same number of topological neighbors, k , and communications are assumed to be perfect. At high density ($\rho \gtrsim 10^{-1}$), the system is able to carry out near-perfect tracking of the target. At these densities, due to the small inter-agent separation and the lack of open space for the target to move into, the agents are able to constantly keep track of the target.

Between these two phases sits a ‘transition’ phase in which the tracking performance, Ξ , sharply rises with ρ . The start of this transition phase can be traced to the ability of agents to effectively cluster around a point of attraction and is important as it allows new agents to takeover the tracking task from the agents that initially encountered the target. This ability to cluster is revealed by the increase in density difference, $\Delta\rho$, (introduced in Sect. 3.4) versus ρ (Fig. 2). The density at which the transition phase starts appears to be independent of N (Fig. 1). The inability of the agents to cluster and make use of target information suggests that at low density, the system’s performance is limited by its lack of exploitation. This is because when the level of connectivity, k , is constant, the same number of agents are drawn to the target. However, when approaching the high density phase, large swarms (i.e., high values of N) outperform smaller ones (i.e., low values of N) since the latter tend to dedicate more resources—i.e., a larger proportion of agents—to exploiting target information, thus leaving very few agents to carry out area exploration. Therefore, it can be said that these small systems tend to have their tracking performance limited by a lack of exploration. In summary, low density swarms are exploitation-limited, while high density ones are exploration-limited. The implications of swarm size and its level of connectivity on the systems EED is explored in the subsequent sections.

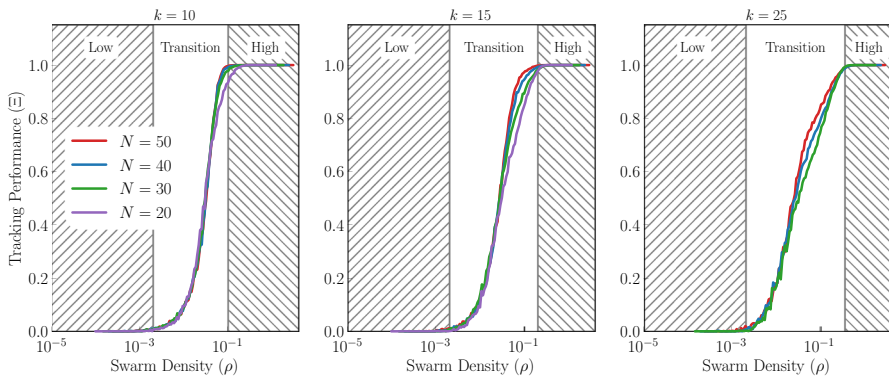


Fig. 1 A swarm of $N = 20, 30, 40,$ and 50 agents with varying levels of connectivity, k , and swarm density, ρ , tracking a fast-moving non-evasive target traveling at $v_{o,max} = 0.15$. In the low density phase, the system is unable to track the target. In the high density phase, the system able to track the target continuously without interruptions. Between these two phases, there is a transition phase in which the tracking performance of systems rapidly increase with swarm density, ρ

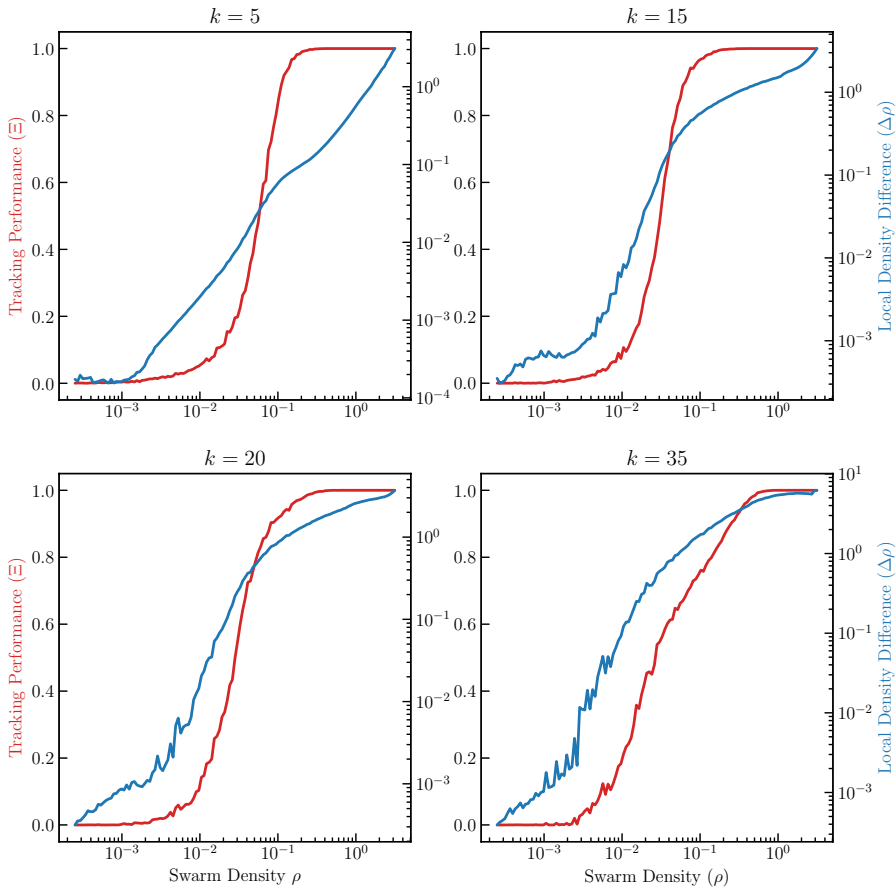


Fig. 2 Density difference, $\Delta\rho$, of a swarm comprised of $N = 50$ agents with various degrees of connectivity, k , tracking a fast-moving non-evasive target traveling at $v_{o,max} = 0.15$

4.2 Effect of swarm density on exploration–exploitation dynamics

To determine the effects of changing a swarm’s density on its exploration–exploitation dynamics (EED), we characterize the system’s EED by calculating an *exploration ratio*, Θ , which is a system-level metric (introduced in Sect. 3.4). By construction, high Θ values indicate that the swarm is biased towards carrying out exploration. Conversely, low Θ values point to a shift of this bias towards exploitation, which may not always consist of ‘useful’ exploitation, as will be discussed later in this section. In addition to quantifying the performance of the swarm, the tracking ability, Ξ , can also be used to quantify the effectiveness of the swarm’s exploitative actions (Sect. 3.4); high levels of Ξ serve as an indicator of a high level of effective exploitation. Note that in what follows, any optimum sought will always be with respect to maximizing Ξ .

The existence of a trade-off between exploration and exploitation for collective tasks has previously been reported in different MAS (Oliveira et al., 2017; Kwa & Bouffanais, 2022). However, a clear understanding of the underpinning of this trade-off is lacking. Here, we provide a quantitative analysis of this EED balance across the density spectrum.

Starting with the low density phase identified in Fig. 1, we observe that the EED balance is fully skewed towards exploration regardless of the level of connectivity, k , (Fig. 3). At such low swarm density, even heavily favoring the conduct of exploitative actions through high levels of k is insufficient to yield minimal tracking of the target. In other words, the swarming strategy is plainly ineffective. As the system moves into the transition phase in Fig. 3, the trade-off between exploration and exploitation becomes apparent. This is rooted in the fact that small inter-agent distances make it easier for individuals to flock together, in turn promoting the exploitation of any target information picked up by exploring agents. As Ξ goes up, more resources become allocated to exploitation, causing the exploration ratio, Θ , to go down in tandem. In this transition phase, the influence of the connectivity, k , is clearly marked. As reported in the previous section, the rise in Ξ occurs at lower density for high k systems compared to low k ones. Specifically, when looking at the $k = 35$ case

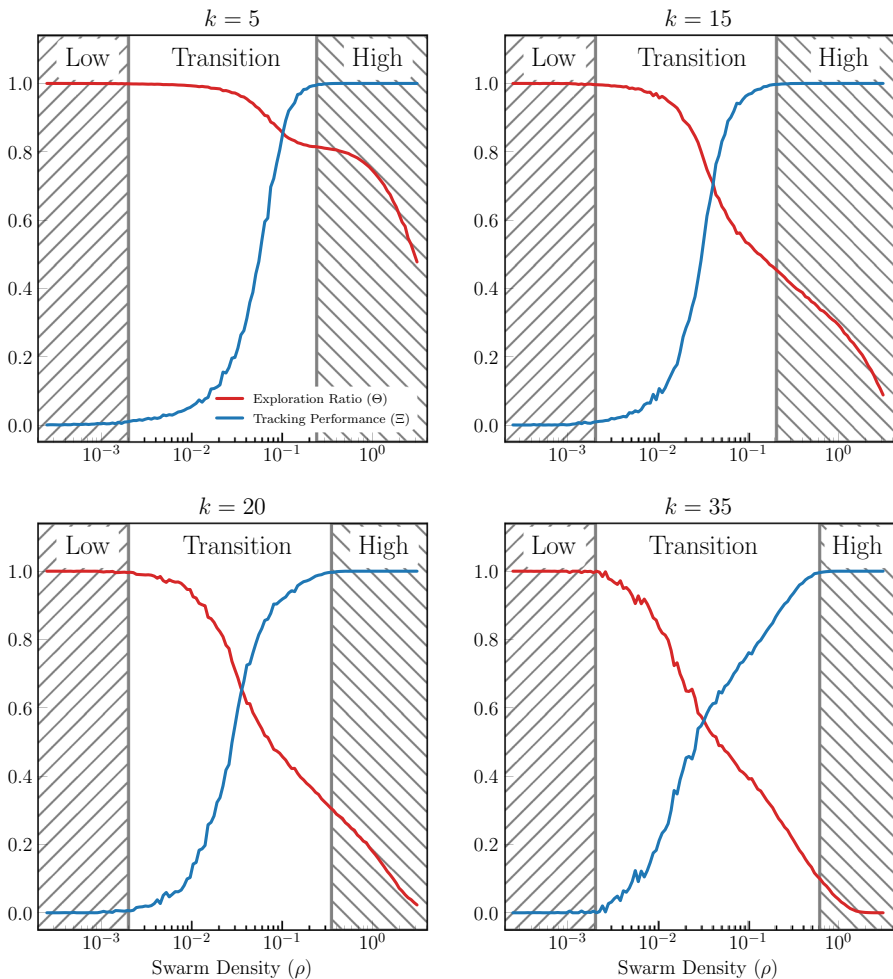


Fig. 3 Tracking performance, Ξ , and exploration ratio, Θ , of a swarm of $N = 50$ agents using with levels of connectivity $k \in [5, 15, 25, 35]$, while tracking a non-evasive target traveling at $v_{o,max} = 0.15$. The low, transition, and high density phases have been marked out on the plots

in Fig. 3, we notice a rapid decrease in Θ across the entire transition phase, almost to the point where no exploration is carried out at all. This should be compared to the case $k = 5$, for which at the higher density end of the transition phase, high Ξ values are achieved while still maintaining a relatively high level of exploration, Θ . It is therefore interesting to notice that in that case, this trade-off is not associated with mutually exclusive actions; it is still possible to maintain high levels of exploration while carrying out similarly high levels of exploitation at intermediate to high densities ($8 \times 10^{-2} \lesssim \rho \lesssim 1 \times 10^0$). This is congruent with the current literature where it has been shown that systems are able to carry out both exploration and exploitation simultaneously (Kwa et al., 2022b). Finally, when moving to the high density phase, which is exploration-limited, we are better able to understand why low k systems outperform high k ones. Indeed, with a low connectivity, maximum tracking performance is achieved while still allocating resources to exploration. This effective EED balance at $k = 5$ therefore helps sustain high levels of system's vigilance, which is key to achieving the coveted high tracking scores. In comparison, with $k = 35$, the rise in Ξ across the transition phase, although starting earlier, is not as rapid as in the case $k = 5$. For $\rho \gtrsim 10^{-1}$ (in the right half of the transition phase), the rise in Ξ markedly slows down, thereby requiring fairly high swarm densities to achieve perfect tracking. This noticeable drop in performance is traced to the reduced system's vigilance given the rapid drop in Θ .

This rich and complex set of behaviors highlights an important fact: adaptivity in the MAS behavior is required when varying the swarm density. Indeed, at a given density, adaptivity allows one to seek the optimal EED balance that yields maximal tracking. Note that in the present search and tracking problem, the level of connectivity k serves as our 'adaptivity' lever. At this stage, the next question arising is how to identify the optimum EED balance (i.e., the value of k here) at any given density ρ .

Since there are different exploration–exploitation balances associated with different permutations of swarm densities and levels of connectivity, there exists an optimum level of connectivity for each density. This is illustrated in Fig. 4a (gray-shaded region) that reveals the existence of a crossover in performance within the transition region. Specifically, at the lower end of the transition phase, a higher level of tracking is achieved with a higher level of connectivity. However, as the density is gradually increased, the converse becomes true.

To gain better insights into these phenomena, we turn to the variations of the exploration ratio, Θ , with ρ . Low degrees of connectivity, k , are associated with higher levels of exploration across the density spectrum (Fig. 4b). Indeed, a reduced connectivity limits the spread of the target's positional information to a small subset of agents. Consequently, the agents beyond that subset are left to explore the surroundings. Since swarms with low k tend to be exploitation-limited, they perform worse when operating at low ρ , where the conditions are known to hamper the conduct of exploitative activities. Conversely, high k values promote the social transmission of information between agents. Thus, agents are better able to exploit this information collaboratively and cluster around the target. This higher level of exploitation allows high k systems to outperform low k systems starting from the lower end of the transition phase (prior to the crossover).

As expected, when increasing the density within the transition phase, the reduced inter-agent distances promote exploitation and the ensuing clustering of agents. Interestingly, at the higher end of the transition phase (past the crossover), low k systems outperform high k ones. Indeed, at these densities, such high k systems exhibit over-exploitative actions, which can be qualified as 'redundant' since excessive agent clustering around the target does not contribute to the tracking performance. As a consequence, these systems at these densities end up being exploration-limited.

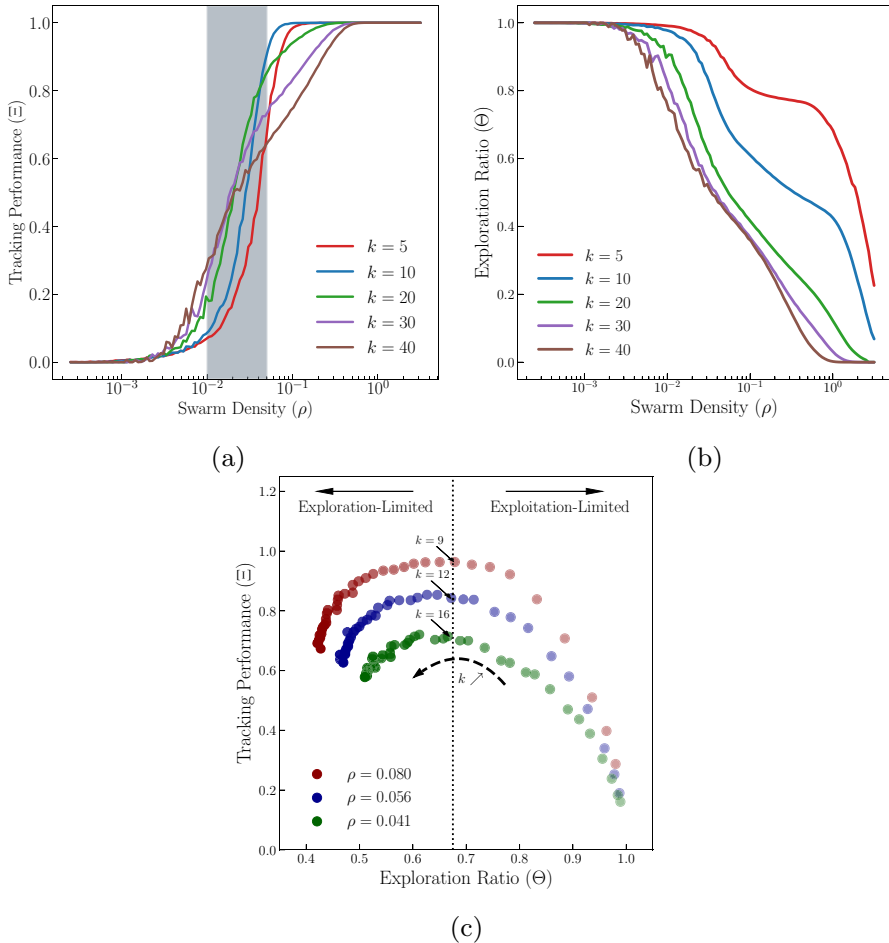


Fig. 4 A swarm of $N = 50$ agents tracking a fast-moving non-evasive target traveling at $v_{o,max} = 0.15$. **a:** Relationship between a swarm’s tracking performance (Ξ) and the average swarm density (ρ) while operating at different levels of connectivity (k). The crossover region, where the optimum k maximizes Ξ , is highlighted in gray. **b:** Relationship between a swarm’s exploration ratio (Θ) and the average swarm density while operating at different levels of connectivity. **c:** Relationship between the system’s tracking performance and exploration ratio. Lighter shaded points represent systems with lower k while darker shaded points represent systems with higher k . The optimum level of exploration, Θ^* , is represented by the dotted line with exploration-limited systems lying on the left of that line, while exploitation-limited systems lie on the right. All three densities considered fall within the transition region (Fig. 1)

The fact that a system’s optimal level of connectivity changes with ρ can be further explained using Fig. 4c that shows the presence of an optimum level of exploration, Θ^* , at which tracking performance is maximized. As already mentioned, for a given swarm density, ρ , there exists an optimal level of connectivity, k^* , such that $\Xi(k^*, \rho) = \max_k(\Xi(\rho))$ (Fig. 4a). In turn, this gives us that $\Theta^* = \Theta(\max_k(\Xi))$. Although this optimum exploration level, Θ^* , cannot easily be deduced from Fig. 4b, it is readily apparent in Fig. 4c (see vertical dotted line). Interestingly, while determining the exact position of Θ^* requires further study, it appears to remain constant even as ρ changes, which is connected to the

changes in k^* from one swarm density to the next. Furthermore, Θ^* clearly delineates between two regions corresponding to exploration-limited and exploitation-limited systems— i.e., systems found to the left of the optimum are limited by their lack of exploration while systems on the right of the optimum have their performances limited by a lack of exploitation. This leads to the important conclusion that a system’s level of connectivity—our ‘adaptivity’ lever—must therefore be adjusted to manage and achieve an optimum balance between exploration and exploitation; a result previously suggested in several studies (Kwa et al., 2020a, 2021; Talamali et al., 2021; Horsevad et al., 2022b).

4.3 Effect of swarm size

In previous sections, we have reported compelling evidence that the concept of swarm density is key to the analysis of any MAS. A minimum density is necessary to trigger effective emergent collective actions. Furthermore, as the density increases and we enter the transition phase, the existence of an intricate balance between exploratory and exploitative actions reveals that these emergent properties are strongly dependent on the density. Nonetheless, one should not put aside unavoidable finite-size effects; for instance, those that arise due to the limited number of agents employed within any MRS, usually constrained by financial and logistical challenges (Schroeder et al., 2019; Horsevad et al., 2022a). As such, we must acknowledge the role that a system’s size plays, and it appears therefore necessary to complement our density analysis by investigating the effects of swarm size.

As previously discussed, in natural animal collectives and computational optimization problems, increasing the number of agents within a system primarily serves to increase the level of exploration carried out. Figure 5 illustrates that this is also the case in the target tracking swarm. Especially as the swarm density increases and when operating at high levels of connectivity, the figure shows that large systems tend to maintain higher exploration ratios compared to small systems. This is because the level of exploration carried out by the swarm varies according to k/N , the number of neighbors an agent has as a ratio to the total size of the swarm, as shown in Fig. 6. These results suggest that exploration is a system-level action and therefore depends on the proportion of agents carrying out exploratory activity. When one agent transitions from exploration to

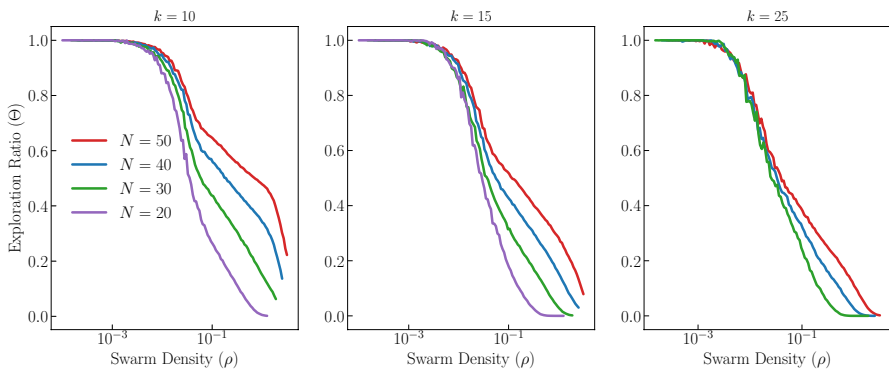


Fig. 5 Exploration ratio of a swarm of $N = 50, 40, 30,$ and 20 agents for three different levels of connectivity, k , and varying swarm densities, ρ , tracking a fast-moving non-evasive target traveling at $v_{o,max} = 0.15$

Fig. 6 Exploration ratio of a swarm of $N = 50, 40, 30,$ and 20 agents with varying levels of connectivity at a swarm density of $\rho = 0.0444$, tracking a fast-moving non-evasive target traveling at $v_{o,max} = 0.15$

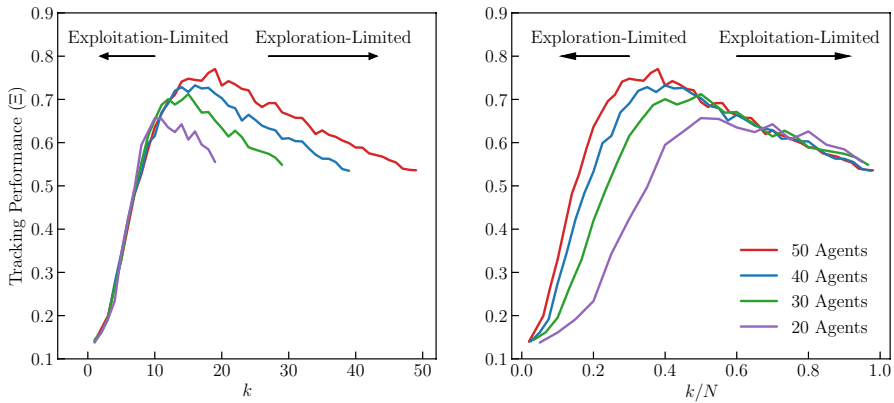
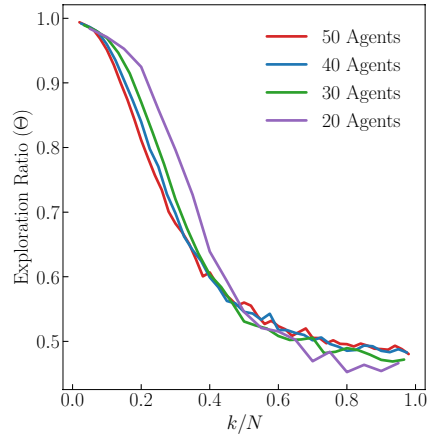


Fig. 7 Tracking performance of a swarm of $N = 50, 40, 30,$ and 20 agents with varying levels of connectivity at a swarm density of $\rho = 0.0444$, tracking a fast-moving non-evasive target traveling at $v_{o,max} = 0.15$. Tracking performances are plot against k (left) and k/N (right)

exploitation, small swarms lose a larger amount of their exploration capacity compared to large swarms (10% in a swarm comprised of 10 agents compared to 2% in a swarm comprised of 50 agents). As such, when using similar levels of connectivity, large systems tend to have a larger proportion of agents that remain unengaged with the target, allowing them to sustain higher levels of exploration compared to systems comprised of fewer units.

The system-level nature of exploration can also be observed when analyzing the tracking performances of differently sized systems. Figure 7 illustrates that there also exists an optimal level of connectivity (k^*) maximizing a system’s tracking performance. At $k < k^*$, the system operates in the exploitation-limited phase. Conversely, at $k > k^*$, the system belongs to the exploration-limited phase. Therein, it can be seen that all systems experience similar losses in performance that are proportional to k/N , regardless of system size. This further highlights that the amount of exploration carried out by a system is effectively a global property.

In contrast, when operating in the exploitation-limited phase, it can be seen that the systems' performances are proportional to k , the absolute number of neighbors, regardless of the system size. This shows that exploitation is a local-level task that depends on the number of agents that are able to cluster around the target. When operating in the exploitation-limited phase, using the same level of connectivity results in the same number of agents being drawn to the target, thereby keeping the level of exploitation constant across systems with different numbers of agents.

Besides the trends observed in exploration and exploitation, Fig. 7 also reveals that small swarms display lower maximum tracking performances, as well as a lower k^* , when compared to a large swarm. This can be traced to the lower capacity of small swarms to concurrently carry out exploratory and exploitative actions. As the level of connectivity of a small swarm rises, the system rapidly dedicates larger proportions of agents to the exploitation task by having them engage with the target. This in turn reduces the proportion of agents carrying out exploration, causing the system to move from the exploitation-limited to the exploration-limited phase more quickly than for larger swarms.

The lack in exploitation capacity also explains the trends seen in Fig. 8 that shows the swarm's performance when it is operating closer to the low density phase. Here, it can be observed that the tracking performance still increases proportionally with k , further confirming that the task of target tracking, and hence exploitation, requires localized coordination. However, for the swarms comprised of 30 and 20 agents, tracking performance continues to rise when k is increased and does not peak. This suggests that, due to the small number of agents and the large distances between them, the swarm is unable to fully exploit the target's positional information when it is found. As such, there is no optimal level of connectivity and no transition to the exploration-limited phase.

5 Discussion

When considering a system of N robots that collaboratively performs a given task over a given domain area, one typically seeks the minimum number of units to accomplish that according to a specific metric. On one hand, with too few robots, the MRS inevitably

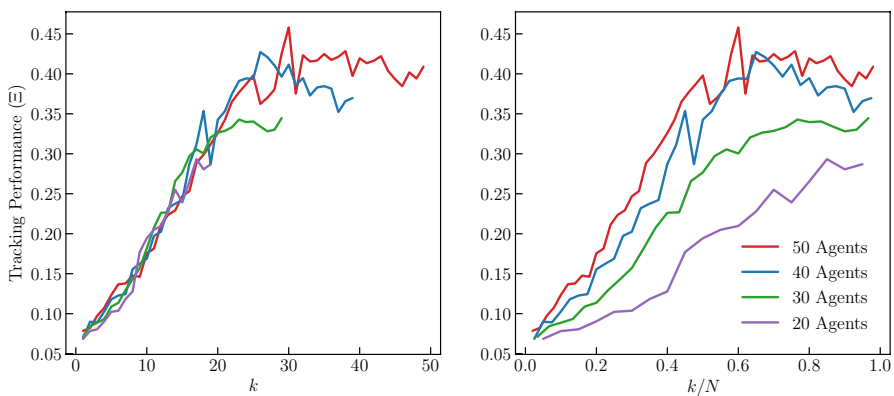


Fig. 8 Tracking performance of a swarm of $N = 50, 40, 30,$ and 20 agents with varying levels of connectivity at a swarm density of $\rho = 0.02$, tracking a fast-moving non-evasive target traveling at $\mathbf{v}_{o,\max} = 0.15$. Tracking performances are plot against k (left) and k/N (right)

fails to coordinate the actions of its individual units, thereby hindering the emergence of a robust collective response. On the other hand, with too many robots, unavoidable interference hampers the coordination of behaviors (Hamann, 2012, 2018b). This points to a range of values for N that is neither too low, nor too high. More generally, this points to the existence of an interval in agents density over which, the system goes from poor (to nonexistent) collective action to a maximum in collective performance.

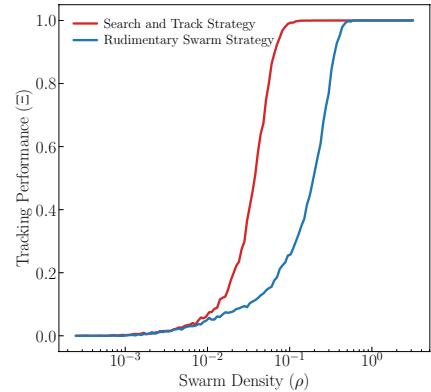
For our collective task of target tracking, we have indeed identified such an interval, with a clear minimum density required to effectively trigger the necessary emergent coordination in the so-called exploitation-limited regime (Fig. 4a). At the other end of this interval, the upper bound in density corresponds to a permanent and sustained tracking of the target by the MRS, within the so-called exploration-limited regime. From the practitioners perspective, one should seek all possible ways to shift this interval towards the lower density values. This would amount to achieving better tracking performance with a smaller system size—i.e., at a reduced overall cost. In other words, advances in the design of the collective actions can help take a fuller advantage of swarm intelligence.

To be specific, the swarm's flexibility is reflected by a particular exploration–exploitation balance, ultimately responsible for the observed variations of the tracking performance with the density. For the sake of this discussion, it is worth recalling the complete definition of flexibility given by Dorigo et al. (2021): “The capacity to solve problems/perform tasks that depart from those chosen at design time.” Needless to say that no absolute metric for flexibility has ever been proposed. We are therefore left to carry out relative comparisons of indirect flexibility measurements that have to be considered across a vast range of operating conditions. Indeed, for a given swarm strategy—i.e., a given set of local update rules—the effectiveness of the collective response of the system when the task/environment demands change is a proxy for flexibility. For our problem of target tracking, flexibility can be probed in many different ways: e.g., by changing the number of targets (Kwa et al., 2021, 2022), the speed of those targets (Kwa et al., 2020a), by adding obstacles (Sun et al., 2022), etc.

As a matter fact, at any given density, the only means to increase the collective tracking performance is to modify this exploration–exploitation balance by identifying potential adaptivity ‘levers’—“the ability to learn/change behavior to respond to new operating conditions” (Dorigo et al., 2021). In the results presented in the previous sections, the only two adaptivity levers involved are: (1) the one associated with the memory introduced in the velocity attraction component (Appendix 1.1), and (2) the adaptive repulsion strategy (Appendix 1.2). This fact is clearly illustrated by Fig. 9, where our search-and-track swarm strategy with $k = 10$ (Sect. 3.1) is shown alongside a strategy, which is identical in all ways except that it is memoryless and has a fixed repulsion strength. As expected, both strategies are equally ineffective at low density. As the swarm density increases, the red tracking performance starts outstripping the blue one, then rapidly increases and peaks at one for a density almost ten times lower than the blue curve. The swarm strategy in red therefore exhibits much higher levels of flexibility than the blue one. For any density within the transition phase, the introduction of the inter-agent adaptive repulsion and memory (Sect. 3.1) taps into adaptivity that yields a more swarm-intelligent collective tracking of the target. Hence, in this context it appears that adaptivity promotes flexibility. It is worth adding that beyond memory and adaptive repulsion, other adaptivity levers exist and can be harnessed to further improve the effectiveness of the collective strategy.

However, for this to be possible, it is key to have a clear understanding of the pivotal role played by the swarm density. As revealing as the previous analysis is, it does not stress enough the critical need to vary the density in order to: (1) identify in which phase the

Fig. 9 Comparison of the search and track strategy presented in Sect. 3.1 and a rudimentary memoryless strategy that uses a constant inter-agent repulsion behavior. Both sets of simulations were carried out at a level of connectivity, $k = 10$. The performance curve for the rudimentary strategy is shifted towards a higher density range compared to the baseline swarm strategy



MRS is effectively operating—low density, transition phase or high density (Fig. 1), and (2) characterize the actual exploration–exploitation balance and the associated level of flexibility. For instance, consider a practitioner designing and operating a given MRS with a given number of units N . Without varying the swarm density, this practitioner would not know whether the system is operating in the exploration-limited or exploitation-limited regime. Should the task require operations over a limited surface area, the density might be too high and swarm intelligence might not be necessary—it could induce interference effects hampering the collective dynamics. Arguably, the most interesting region is the transition phase, where adaptivity can be sought to boost swarm intelligence and deliver higher performances at the same density, or equivalently offer the same performance with a much lower number of units N . Beyond the need to identify the density interval corresponding to the transition phase, one also has to determine whether at that given density, the MRS behavior is exploration-limited or exploitation-limited. Failure to do so would significantly complicate the search for an effective adaptivity strategy. As is seen in Fig. 4a, in the lower range of densities within the transition phase, one can make use of the system’s level of connectivity as another adaptivity lever and increase k . However, this strategy completely backfires in the exploration-limited regime for slightly higher densities, still in the transition phase.

The previous analysis about the density dependence of robot swarms also helps explain some aspects of the simulation–reality gap for MRS. The latter has been acknowledged to be particularly wide owing to its typical characteristics as a complex system—i.e., the large number of interacting units responsible for the emergent behaviors (Francesca & Birattari, 2016; Dorigo et al., 2021). Indeed, challenges in adequately and accurately modeling and/or simulating robot swarms can lead to non-negligible differences in swarm density, which in turn would induce notable discrepancies between simulations and hardware experiments (Hamann, 2018b; Zhong et al., 2018; Dorigo et al., 2021; Kwa et al., 2020a, 2021).

It is important to stress that the obtained results and analysis have been established for our problem of collective tracking. The question of how these results for this particular collective action can be generalized to other swarming behaviors naturally arises. At this point, it is worth recalling that few general results related to swarming behaviors effectively exist; there is no general scalability law, no robustness law, or flexibility law for that matter. The emergent nature of the collective action and the fact that swarms are complex systems can explain this dearth of general results, which is further compounded by the current lack

of benchmark problems to quantitatively assess and compare the overall performance of a system with a given cooperative control algorithm (Kwa et al., 2022b). Nonetheless, the key question about the influence of the swarm density on the emergent response of MRS remains applicable and valid for all robot swarms involved in a wide range of collective actions. As mentioned earlier, at extremely low density the swarm inevitably fails to self-organize. It is only by increasing the number of units—and thus the swarm density—can an emergent behavior materialize. At the other end of the density spectrum, a high swarm density means that so many units are available that swarm intelligence may not even be necessary to carry out the task at hand—too high a density may even hinder the collective organization. These well-known facts are unambiguous about the existence of a transition phase in which the system dynamics goes from lacking self-organization to exhibiting an emergent response. This transition phase has been observed in a number of collective behaviors including with Vicsek’s model for self-propelled particles (Vicsek et al., 1995; Bouffanais, 2016). Although the existence of this transition phase is not fundamentally new, its implications for the design of robot swarms performing any task—beyond just collective tracking—has been overlooked, in particular in relation with the sizing of the system. When characterizing the transition phase, some differences should naturally be expected across a wide range of tasks and problems. For instance, the adverse effects of interference at high density can effectively induce a drop in performance as has been recognized for foraging tasks with a central depository (Rosenfeld et al., 2006; Hamann, 2012). Nevertheless, a systematic analysis of this transition phase for the class of collective tasks involving an exploration–exploitation balance (Kwa et al., 2022b) would contribute to further generalizing some of the results reported here. We therefore hope that this work will encourage other researchers to study the influence of swarm density across a wider range of platforms and tasks in swarm robotics.

We can conclude that an analysis of the density dependence for an MRS offers means of comparing and quantifying swarm intelligence when it is engaged in a particular task. This is a necessary step towards optimizing the collective performance, i.e., when seeking to boost swarm intelligence. From the machine learning perspective, these results and conclusions have far-reaching implications. Numerous strategies have been considered to establish the local control rules that lead to sought-after global collective actions: e.g., automatic design (Francesca & Birattari, 2016; Ligot et al., 2022), multi-agent reinforcement learning (Kouzehgar et al., 2020; Zhang et al., 2021), deep reinforcement learning (Hüttenrauch et al., 2019), etc. However, in most of these approaches, the number of units (or equivalently the swarm density) is a parameter that is considered constant during the learning/optimization process. Based on the above analysis of the density dependence, one has to be cautious about the fact that if the learning/optimization process is simply carried out at a given density, it has the potential to yield dismal performance should the circumstances suddenly change. For instance, consider a learned behavior for a density in the upper range of the transition phase (say $\rho \simeq 10^{-1}$ in Fig. 4a), which should offer a relatively high tracking performance. Given what is found in Sect. 4, the learned behavior will most likely find ways to promote exploration, which limits the tracking at that density. Should the density suddenly drop due to the loss of units or an expansion of the domain area, it is highly probable that the learned behavior optimized for exploration will under-perform given the increased need for exploitation. Similarly, the identified influence of swarm size in the exploration-limited regime (Sect. 4.3) should also be taken into consideration when resorting to machine learning approaches.

This does not imply that machine learning strategies should not be considered for this class of problems. Instead, what our analysis suggests is to use machine learning strategies

that aim at optimizing adaptivity across the density spectrum. In conclusion, one should aim towards “adaptivity learning” for MRS, while avoiding the learning of a specific set of behaviors for a given task and circumstances. To help convey the deep meaning of adaptivity learning, we can return once again to Fig. 4a, and learning adaptivity would amount to finding the optimal value of k that yields the highest tracking performance for any value of the swarm density. As mentioned earlier, with our search and track strategy (Sect. 3.1), another potential adaptivity ‘lever’—beyond memory and the repulsive behavior—is the system’s connectivity level, k . A caveat with this adaptivity learning approach is that it does not seem trivial to generalize to any swarm strategy. Nonetheless, the identification of all adaptivity levers associated with any swarm strategy is a first step in that direction.

Lastly, it can be argued that learning adaptivity is equivalent to dynamically optimizing the balance between exploitative and exploratory actions at the system level when circumstances and/or operating conditions change. We believe that learning adaptivity and the optimization of the exploration–exploitation balance will offer promising new opportunities and avenues for the community working on the design of effective collective swarm dynamics.

Appendix 1 Strategy velocity components

The search and track strategy given in Sect. 3.1 produces a velocity vector comprised of two parts: (1) the attraction velocity component, $\mathbf{v}_{i,\text{att}}[t]$, and (2) the repulsion velocity component, $\mathbf{v}_{i,\text{rep}}[t]$. These two components are then combined to give a final agent velocity using Eq. 1, which is restated here:

$$\mathbf{v}_i[t] = \mathbf{v}_{i,\text{att}}[t] + \mathbf{v}_{i,\text{rep}}[t]. \quad (7)$$

In this section, we state how the values for $\mathbf{v}_{i,\text{att}}[t]$ and $\mathbf{v}_{i,\text{rep}}[t]$ are obtained. This strategy was first presented in Kwa et al. (2022).

1.1 Attraction velocity component

The attractive component is used to encourage agents to aggregate at a point of interest, $\mathbf{p}[t]$, determined using Algorithm 2. At every time-step, each agent measures its local environment to look for a target. Should an agent detect a target, the agent will transition from an exploratory state into a tracking state, set $\mathbf{p}[t]$ as the target’s current location, and broadcast the location. Should a target not be detected, the agent will communicate with its k -nearest neighbors and attempt to track targets detected by its neighbors. In addition, each agent is endowed with a memory, M , of a duration of t_{mem} . Using this memory, each agent is able to keep track of the position and time at which a target was found. Each agent also receives a set of target positions and encounter times from its k -nearest neighbors. These received values are compared to an agent’s own values and the most recent target position is used as a point of attraction, $\mathbf{p}[t]$. At this point, should the agent still not have any knowledge of the target’s location, $\mathbf{p}[t]$ is set to the agent’s own location, $x_i[t]$, essentially disabling the attractive component. Through the use of this update algorithm, agents are able to compare information that is directly sensed from the environment with information received from its neighbors and choose which set of information to exploit.

At this point, it is important to reemphasize that in this framework, the neighborhood of an agent is to be understood in the network sense. As such, an agent i has as many neighbors as its degree, k . Also, since time-varying network topologies are considered, it should be noted the neighborhood of each agent evolves over the course of the task duration. Given this dynamic network topology, all agents independently set $\mathbf{p}[t]$ using Algorithm 2.

Algorithm 2 : Point of Attraction Update Algorithm

```

Initialise  $M = t_{\text{mem}}$ 
if Agent detects target then
     $\mathbf{p}_{\text{self}} \leftarrow$  Target's position
     $t_{\text{best}} \leftarrow t$ 
end if
Determine  $\mathcal{N}_i = \{j \in [1, N] \text{ s.t. agent } j \text{ is a topological } k\text{-nearest neighbor of agent } i\}$ 
Get list of all neighbors'  $\mathbf{p}$  and  $t_{\text{best}}$ 
 $\mathbf{p}_{\text{neigh}} \leftarrow$  Most recent entry in all neighbors'  $\mathbf{p}$ 
 $t_{\text{neigh}} \leftarrow$  Most recent entry in all neighbors'  $t_{\text{best}}$ 
if  $t_{\text{best}} + M < t$  then
     $\mathbf{p}_{\text{self}} \leftarrow \emptyset$ 
end if
if  $t_{\text{neigh}} + M < t$  then
     $\mathbf{p}_{\text{neigh}} \leftarrow \emptyset$ 
end if
if  $\mathbf{p}_{\text{self}} = \emptyset$  and  $\mathbf{p}_{\text{neigh}} = \emptyset$  then
     $\mathbf{p}[t] \leftarrow \mathbf{x}_i[t]$ 
else if  $t_{\text{best}} > t_{\text{neigh}}$  then
     $\mathbf{p}[t] \leftarrow \mathbf{p}_{\text{self}}$ 
else
     $\mathbf{p}[t] \leftarrow \mathbf{p}_{\text{neigh}}$ 
end if

```

Using an agent's velocity in the previous time-step and its location in relation to $\mathbf{p}[t]$, the attraction component can be calculated according to:

$$\mathbf{v}_{i,\text{att}}[t] = \omega \mathbf{v}_i[t-1] + cr(\mathbf{p}[t] - \mathbf{x}_i[t]). \quad (8)$$

This equation is similar to that used in the social-only PSO model proposed by Engelbrecht (2010), where ω is the velocity inertial weight, c is the social weight, and r is a number randomly drawn from the unit interval. In computational optimization, this is the main driver of a the system's exploitative behaviour. Here, it is used to drive the MRS towards the target. It should be noted that in the proposed strategy, unlike the social-only PSO model that uses an infinite memory length, agents here instead use a limited memory length. This is done to prevent agents from exploiting outdated target positional information.

1.2 Adaptive repulsion

The adaptive repulsion component is used to promote agent exploration of the search space and stop agents from aggregating within a small area, thereby preventing over-exploitation of target information. In addition, this behavior also offers an anti-collision measure as a direct byproduct of this mechanism.

The inter-agent repulsion scheme adopted is based on the one used in the BoB swarm developed by Vallegra et al. (2018); Zoss et al. (2018). Using this behavior, an Agent i with topological neighbors j calculates its individual repulsion velocity as follows:

$$\mathbf{v}_{i,\text{rep}}[t] = - \sum_{j \in \mathcal{N}_i} \left(\frac{a_R[t]}{r_{ij}[t]} \right)^d \frac{\mathbf{r}_{ij}[t]}{r_{ij}[t]}, \quad (9)$$

where \mathbf{r}_{ij} is the vector from Agent i to Agent j and $r_{ij} = \|\mathbf{r}_{ij}\|$. This inter-agent repulsion is controlled by two parameters: the repulsion strength a_R , affecting the agents' distance from each other at equilibrium, and the exponent d in the pre-factor term (a_R/r_{ij}). In the work carried out, d is fixed at 6 given that this value has very moderate effects on the performance of the EED strategy. At large (a_R/r_{ij}) and d values, the repulsion strength of the agents is approximately equal to the nearest-neighbor distance in equilibrium configuration (Vallegra et al., 2018; Coquet et al., 2021).

The key aspect of this inter-agent repulsion is an agent's ability to adjust its own repulsion strength, $a_R[t]$, based on its local environment and neighbourhood. To this end, the agent's exploratory state, $S_{i,\text{exp}}[t]$, is used. When an agent has no target information, it enters an exploratory state, i.e., $S_{i,\text{exp}}[t] = 1$, it increases its a_R value until a maximum value is attained. Conversely, if the agent is in a tracking state, i.e., $S_{i,\text{exp}}[t] = 0$, the agent gradually reduces its a_R value until a minimum value is reached. The adaptive repulsion behavior used to obtain the repulsion component is summarized in Algorithm 3.

Algorithm 3 Adaptive Repulsion

```

Initialise  $a_{R,\text{min}}$ ,  $a_{R,\text{max}}$ ,  $d = 6$ ,  $\delta_{\text{explore}} = 0.1$ , and  $\delta_{\text{track}} = 0.75$ 
while System active do
  if  $S_{i,\text{exp}}[t] = 0$  then // Agent in tracking state. Reduce  $a_R$ .
    if  $a_R > a_{R,\text{min}}$  then
       $a_R \leftarrow a_R - \delta_{\text{track}}$ 
    end if
  else if  $a_R < a_{R,\text{max}}$  then // Agent in exploratory state. Increase  $a_R$ .
     $a_R \leftarrow a_R + \delta_{\text{explore}}$ 
  end if
  Calculate  $\mathbf{v}_{i,\text{rep}}$  using Eq. A3
end while

```

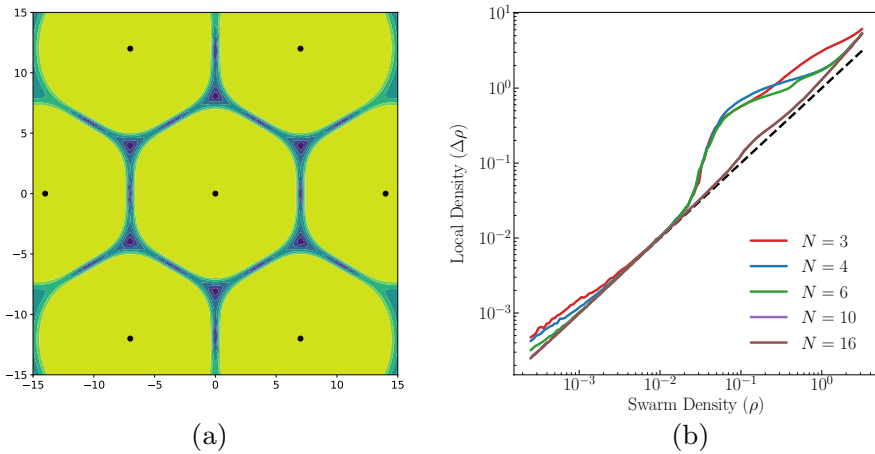


Fig. 10 a: Positioning of agents (black dots) in relation to each other and the repulsion fields generated by the individual agents while in their equilibrium positions. Areas in yellow represent areas of high repulsion potential while those in blue represent areas of low repulsion potential. Given these repulsion potential fields, agents tend to fall into a hexagonal packing pattern around each other. **b:** Local agent density calculated with different number of neighboring agents for a swarm comprised of 50 agents, connected using a $k = 20$ communications network. The system is tracking a non-evasive target traveling at a maximum speed of $v_{o,max} = 0.15$. The local agent density is compared against the global average swarm density (dashed line) (Color figure online)

Appendix 2 Local density

This section shows the reasoning why Eq. 6 was used in the calculation of the system’s local swarm density. This is restated here for completeness:

$$\rho_L = \frac{1}{NT} \sum_{t=1}^T \sum_{i=1}^N \frac{7}{\pi L_{i,t}^2}. \tag{10}$$

Due to the implemented inter-agent repulsion behavior, agents will tend to fall into a hexagonal packing pattern as seen in Fig. 10a. As such, an individual agent will usually be surrounded by six other neighboring agents unless they are located at the edges of the system. By defining an L_i as the average distance between an agent i and its 6 nearest neighbors, it can be assumed that all 6 neighboring agents are located a distance of L_i away for the purposes of calculation of an individual agent’s local agent density.

While a different number of agents can be used for this calculation, the same trends in the local agent density when varying the global average swarm density, as seen in Fig. 10b. However, if less agents are used in this calculation, the initial divergence between the local agent density and the global average swarm density is accentuated. As such, an agent that finds itself in close proximity (relative to the size of the environment) to another agent while moving around the domain will return an artificially high local density. Conversely, if too many agents are used, the presence of such coincidental agent ‘clusters’ is not reflected. As such, an intermediate number, six in this case, was chosen to be used for the local density calculations.

Acknowledgements Not applicable

Author Contributions Conceptualization: RB; Methodology: HLK & RB; Development of Simulation and Data Processing Tools: HLK; Conduct of Experiments: HLK; Data Analysis: HLK, JP & RB; Manuscript Preparation and Review: HLK, JP & RB

Funding This work was supported by the Thales Solutions Asia under the Singapore Economic Development Board Industrial Postgraduate Programme (EDB IPP) and the Natural Sciences and Engineering Research Council of Canada (NSERC), under the grant # RGPIN-2022-04064.

Data availability The data used in this study can be found in the following GitHub repository <https://github.com/hianlee/swarm-density-tracking>.

Declarations

Conflict of interest H. L. Kwa is employed as a Research Engineer and receives a salary from Thales Solutions Asia. All other authors have no relevant financial or non-financial interests to disclose.

Consent for publication Not applicable

Ethical approval Not applicable

References

- Açikmeşe, B., & Bayard, D.S. (2012). A markov chain approach to probabilistic swarm guidance. In 2012 American Control Conference (ACC). IEEE, (pp 6300–6307). Montreal, <https://doi.org/10.1109/ACC.2012.6314729>
- Biswal, S., Elamvazhuthi, K., & Berman, S. (2021). Decentralized control of multi-agent systems using local density feedback. *IEEE Transactions on Automatic Control*, 67(8), 3920–3932. <https://doi.org/10.1109/TAC.2021.3109520>
- Bouffanais, R. (2016). *Design and control of swarm dynamics*. Singapore: Springer. <https://doi.org/10.1007/978-981-287-751-2>
- Cates, M. E., & Tailleur, J. (2015). Motility-induced phase separation. *Annual Review of Condensed Matter Physics*, 6(1), 219–244. <https://doi.org/10.1146/annurev-conmatphys-031214-014710>
- Coquet, C., Aubry, C., & Arnold, A., et al. (2019). A local charged particle swarm optimization to track an underwater mobile source. In OCEANS 2019 - Marseille. IEEE, Marseille. <https://doi.org/10.1109/OCEANSE.2019.8867527>
- Coquet, C., Arnold, A., & Bouvet, P. J. (2021). Control of a robotic swarm formation to track a dynamic target with communication constraints: Analysis and simulation. *Applied Sciences*, 11(7). <https://doi.org/10.3390/app11073179>
- Crosscombe, M., & Lawry, J. (2021). The impact of network connectivity on collective learning. In *Proceedings of the 15th International Symposium on Distributed Autonomous Robotics Systems (DARS21)*. https://doi.org/10.1007/978-3-030-92790-5_7
- Dadgar, M., Couceiro, M. S., & Hamzeh, A. (2017). RDPSO diversity enhancement based on repulsion between similar ions for robotic target searching. In 2017 Artificial Intelligence and Signal Processing Conference (AISP). (pp 275–280). Shiraz <https://doi.org/10.1109/AISP.2017.8324096>
- Dadgar, M., Couceiro, M. S., & Hamzeh, A. (2020). RbRDPSO: Repulsion-based RDPSO for robotic target searching. *Iranian Journal of Science and Technology - Transactions of Electrical Engineering*, 44(1), 551–563. <https://doi.org/10.1007/s40998-019-00245-z>
- de Souza, C., Castillo, P., & Vidolov, B. (2022). Local interaction and navigation guidance for hunters drones : A chase behavior approach with real-time tests. *Robotica*, 40(8), 1–19. <https://doi.org/10.1017/S0263574721001910>
- Dorigo, M., Theraulaz, G., & Trianni, V. (2021). Swarm robotics: Past, present, and future. *Proceedings of the IEEE*, 109(7), 1152–1165. <https://doi.org/10.1109/jproc.2021.3072740>

- Ebert, J. T., Gauci, M., & Nagpal, R. (2018). Multi-feature collective decision making in robot swarms. In: Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS. (pp. 1711–1719). Stockholm. <https://doi.org/10.5555/3237383.3237953>
- Ebert, J. T., Gauci, M., & Mallmann-trenn, F., et al. (2020). Bayes Bots : Collective bayesian decision-making in decentralized robot swarms. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). (pp. 7186–7192) Paris. <https://doi.org/10.1109/ICRA40945.2020.9196584>
- Elamvazhuthi, K., & Berman, S. (2019). Mean-field models in swarm robotics: A survey. *Bioinspiration & Biomimetics*. <https://doi.org/10.1088/1748-3190/ab49a4>
- Engelbrecht, A. P. (2010). Heterogeneous particle swarm optimization. In: Dorigo, M. et al (ed) 7th Int. Conf. ANTS 2010. (pp. 191–202). Springer: Berlin,
- Esterle, L., & Lewis, P. R. (2020). Distributed autonomy and trade-offs in online multiobject k-coverage. *Computational Intelligence*, 36(2), 720–742. <https://doi.org/10.1111/coin.12264>
- Francesca, G., & Birattari, M. (2016). Automatic design of robot swarms: achievements and challenges. *Frontiers in Robotics and AI*, 3, 29.
- Hamann, H. (2012). Towards swarm calculus: Universal properties of swarm performance and collective decisions. In: Swarm Intelligence: 8th International Conference, ANTS 2012. Volume 7461 of LNCS. (pp 168–179). Springer: Berlin
- Hamann, H. (2018a). Superlinear scalability in parallel computing and multi-robot systems: Shared resources, collaboration, and network topology. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 10793, 31–42. https://doi.org/10.1007/978-3-319-77610-1_3
- Hamann, H. (2018). *Swarm robotics: A formal approach*. London: Springer International Publishing. <https://doi.org/10.1007/978-3-319-74528-2>
- Hecker, J. P., & Moses, M. E. (2015). Beyond pheromones: evolving error-tolerant, flexible, and scalable ant-inspired robot swarms. *Swarm Intelligence*, 9(1), 43–70. <https://doi.org/10.1007/s11721-015-0104-z>
- Hönig, W., & Ayanian, N. (2016). Dynamic multi-target coverage with robotic cameras. In IEEE International Conference on Intelligent Robots and Systems (pp. 1871–1878). Daejeon, <https://doi.org/10.1109/IROS.2016.7759297>
- Hornischer, H., Varughese, J. C., Thenius, R., et al. (2020). CIMAX: Collective information maximization in robotic swarms using local communication. *Adaptive Behavior*, 29(3). <https://doi.org/10.1177/1059712320912021>
- Horsevad, N., Kwa, H. L., & Bouffanais, R. (2022a). Beyond bio-inspired robotics: How multi-robot systems can support research on collective animal behavior. *Frontiers in Robotics and AI*. <https://doi.org/10.3389/frobt.2022.865414>
- Horsevad, N., Mateo, D., Kooij, R. E., et al. (2022b). Transition from simple to complex contagion in collective decision-making. *Nature Communications*, 13, 1442. <https://doi.org/10.1038/s41467-022-28958-6>
- Hüttenrauch, M., Adrian, S., Neumann, G., et al. (2019). Deep reinforcement learning for swarm systems. *Journal of Machine Learning Research*, 20(54), 1–31.
- Jensen, E. A., Lowmanstone, L., & Gini, M. (2018). Communication-restricted exploration for search teams. *Distributed Autonomous Robotic Systems Springer Proceedings in Advanced Robotics*, 6, 17–30. <https://doi.org/10.1007/978-3-319-73008-0>
- Jurt, M., Milner, E., Sooriyabandara, M., et al. (2022). Collective transport of arbitrarily shaped objects using robot swarms. *Artificial Life and Robotics*. <https://doi.org/10.1007/s10015-022-00730-5>
- Khaluf, Y., Birattari, M., & Rammig, F. (2013). Probabilistic analysis of long-term swarm performance under spatial interferences. In International Conference on Theory and Practice of Natural Computing (pp. 121–132). Caceres. https://doi.org/10.1007/978-3-642-45008-2_10
- Khaluf, Y., Pinciroli, C., Valentini, G., et al. (2017). The impact of agent density on scalability in collective systems: Noise-induced versus majority-based bistability. *Swarm Intelligence*, 11(2), 155–179. <https://doi.org/10.1007/s11721-017-0137-6>
- Kit, J. L., Dharmawan, A. G., & Mateo, D., (2019). Decentralized multi-floor exploration by a swarm of miniature robots teaming with wall-climbing units. In International Symposium on Multi-Robot and Multi-Agent Systems (MRS). IEEE, New Brunswick. <https://doi.org/10.1109/MRS.2019.8901058>
- Kouzehgar, M., Meghjani, M., & Bouffanais, R. (2020). Multi-agent reinforcement learning for dynamic ocean monitoring by a swarm of buoys. In: IEEE-MTS Global Oceans 2020: Singapore–US Gulf Coast, IEEE, pp 1–8, <https://doi.org/10.1109/IEEECONF38699.2020.9389128>
- Kwa, H. L., & Bouffanais, R. (2022). The effect of network connectivity on exploration and exploitation during decentralized collective learning. In 2022 International Workshop on Agent-Based Modelling of Human Behaviour (ABMHuB), Online, http://abmhub.cs.ucl.ac.uk/2022/camera_ready/Kwa_Bouffanais.pdf

- Kwa, H. L., Kit, J. L., & Bouffanais, R. (2020a). Optimal swarm strategy for dynamic target search and tracking. In: Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), Auckland, New Zealand, pp 672–680, <https://doi.org/10.5555/3398761.3398842>
- Kwa, H.L., Tokić, G., & Bouffanais, R., et al (2020b). Heterogeneous swarms for maritime dynamic target search and tracking. In Global OCEANS 2020: Singapore-U.S Gulf Coast. IEEE, Singapore, <https://doi.org/10.1109/IEEECONF38699.2020.9389145>
- Kwa, H. L., Kit, J. L., & Bouffanais, R. (2021). Tracking multiple fast targets with swarms : Interplay between social interaction and agent memory. In: ALIFE 2021: The 2021 Conference on Artificial Life, Prague, Czech Republic, https://doi.org/10.1162/isal_a_00376
- Kwa, H. L., Babineau, V., Philippot, J., et al. (2022). Adapting the exploration-exploitation balance in heterogeneous swarms: Tracking evasive targets. *Artificial Life*, 29, 1–16. https://doi.org/10.1162/artl_a_00390
- Kwa, H. L., Kit, J. L., & Bouffanais, R. (2022). Balancing collective exploration and exploitation in multi-agent and multi-robot systems: A review. *Frontiers in Robotics and AI*. <https://doi.org/10.3389/frobt.2021.771520>
- Lerman, K., & Galstyan, A. (2001). Mathematical model of foraging in a group of robots: Effect of interference. *Autonomous Robots*, 13(2), 127–141. <https://doi.org/10.1023/A:1019633424543>
- Li, H., Feng, C., & Ehrhard, H., et al. (2017). Decentralized stochastic control of robotic swarm density: Theory, simulation, and experiment. In: IEEE International Conference on Intelligent Robots and Systems, (pp. 4341–4347). Vancouver. <https://doi.org/10.1109/IROS.2017.8206299>
- Ligot, A., Cotorruelo, A., Garone, E., et al. (2022). Towards an empirical practice in off-line fully-automatic design of robot swarms. *IEEE Transactions on Evolutionary Computation*. <https://doi.org/10.1109/TEVC.2022.3144848>
- Liu, Z., Crosscombe, M., & Lawry, J. (2021). Imprecise fusion operators for collective learning. In: ALIFE 2021: The 2021 Conference on Artificial Life, https://doi.org/10.1162/isal_a_00407
- Mateo, D., Kuan, Y. K., & Bouffanais, R. (2017). Effect of correlations in swarms on collective response. *Scientific Reports*. <https://doi.org/10.1038/s41598-017-09830-w>
- Mateo, D., Horsevad, N., Hassani, V., et al. (2019). Optimal network topology for responsive collective behavior. *Science Advances*, 5(4), eaau0999. <https://doi.org/10.1126/sciadv.aau099>
- Oliveira, M., Pinheiro, D., & Macedo, M., et al. (2017). Better exploration-exploitation pace, better swarm: Examining the social interactions. In 2017 IEEE Latin American Conference on Computational Intelligence (LA-CCI). IEEE, Arequipa, Peru, <https://doi.org/10.1109/LA-CCI.2017.8285712>
- Pang, B., Song, Y., Zhang, C., et al. (2019). A swarm robotic exploration strategy based on an improved random walk method. *Journal of Robotics*. <https://doi.org/10.1155/2019/6914212>
- Piotrowski, A. P., Napiorkowski, J. J., & Piotrowska, A. E. (2020). Population size in particle swarm optimization. *Swarm and Evolutionary Computation*, 58(100), 718. <https://doi.org/10.1016/j.swevo.2020.100718>
- Prasetyo, J., De Masi, G., & Ferrante, E. (2019). Collective decision making in dynamic environments. *Swarm Intelligence*, 13, 217–243. <https://doi.org/10.1007/s11721-019-00169-8>
- Rausch, I., Reina, A., Simoens, P., et al. (2019). Coherent collective behaviour emerging from decentralised balancing of social feedback and noise. *Swarm Intelligence*, 13, 321–345. <https://doi.org/10.1007/s11721-019-00173-y>
- Roeva, O., Fidanova, S., & Paprzycki, M. (2015). Population size influence on the genetic and ant algorithms performance in case of cultivation process modeling. *Recent Advances in Computational Optimization*. https://doi.org/10.1007/978-3-319-12631-9_7
- Rosenfeld, A., Kaminka, G. A., & Kraus, S. (2006). A study of scalability properties in robotic teams. In P. Scerri, R. Vincent, & R. Mailler (Eds.), *Coordination of Large-Scale Multiagent Systems* (pp. 27–51). Boston: Springer.
- Rossides, G., Metcalfe, B., & Hunter, A. (2021). Particle swarm optimization—An adaptation for the control of robotic swarms. *Robotics*, 10(2), 58. <https://doi.org/10.3390/robotics10020058>
- Rubenstein, M., Ahler, C., Hoff, N., et al. (2014). Kilobot: A low cost robot with scalable operations designed for collective behaviors. *Robotics and Autonomous Systems*, 62(7), 966–975. <https://doi.org/10.1016/j.robot.2013.08.006>
- Schaerf, T. M., Makinson, J. C., Myerscough, M. R., et al. (2013). Do small swarms have an advantage when house hunting? The effect of swarm size on nest-site selection by *apis mellifera*. *Journal of the Royal Society Interface*. <https://doi.org/10.1098/rsif.2013.0533>

- Schranz, M., Di Caro, G. A., Schmickl, T., et al. (2021). Swarm intelligence and cyber-physical systems: Concepts, challenges and future trends. *Swarm and Evolutionary Computation*, 60, 100762. <https://doi.org/10.1016/j.swevo.2020.100762>
- Schroeder, A., Trease, B., & Arsie, A. (2019). Balancing robot swarm cost and interference effects by varying robot quantity and size. *Swarm Intelligence*, 13(1), 1–19. <https://doi.org/10.1007/s11721-018-0161-1>
- Sekunda, A., Komareji, M., & Bouffanais, R. (2016). Interplay between signaling network design and swarm dynamics. *Network Science*, 4(2), 244–265. <https://doi.org/10.1017/nws.2016.5>
- Shishika, D., & Paley, D. A. (2019). Mosquito-inspired distributed swarming and pursuit for cooperative defense against fast intruders. *Autonomous Robots*, 43(7), 1781–1799. <https://doi.org/10.1007/s10514-018-09827-y>
- Strickland, L., Baudier, K., & Bowers, K., et al. (2018). Bio-inspired role allocation of heterogeneous teams in a site defense task. In: Distributed Autonomous Robotic Systems 2018. Springer International Publishing, Boulder, CO, USA, https://doi.org/10.1007/978-3-030-05816-6_10
- Sun, Z., Sun, H., Li, P., et al. (2022). Self-organizing cooperative pursuit strategy for multi-usv with dynamic obstacle ships. *Journal of Marine Science and Engineering*. <https://doi.org/10.3390/jmse10050562>
- Sung, Y., Budhiraja, A.K., & Williams, R.K., et al. (2018). Distributed simultaneous action and target assignment for multi-robot multi-target tracking. In: Proceedings - IEEE International Conference on Robotics and Automation IEEE, (pp. 3724–3729). Brisbane. <https://doi.org/10.1109/ICRA.2018.8460974>
- Sung, Y., Budhiraja, A. K., Williams, R. K., et al. (2020). Distributed assignment with limited communication for multi-robot multi-target tracking. *Autonomous Robots*, 44, 57–73. <https://doi.org/10.1007/s10514-019-09856-1>
- Talamali, Mohamed S., Saha, Arindam, Marshall, James A. R., & Reina, Andreagiovanni. (2021). When less is more: Robot swarms adapt better to changes with constrained communication. *Science Robotics*, 6(56). <https://doi.org/10.1126/scirobotics.abf1416>
- Thenius, R., Moser, D., Varughese, J. C., et al. (2016). subCULTron - cultural development as a tool in underwater robotics. *Artificial Life and Intelligent Agents*, 732, 27–41. https://doi.org/10.1007/978-3-319-90418-4_3
- Vallegra, F., Mateo, D., & Tokić, G., et al. (2018). Gradual Collective Upgrade of a Swarm of Autonomous Buoys for Dynamic Ocean Monitoring. In IEEE-MTS OCEANS 2018, Charleston, SC, USA, <https://doi.org/10.1109/OCEANS.2018.8604642>
- D Van Den Bergh, F., & Engelbrecht, A.P. (2001). Effects of swarm size on cooperative particle swarm optimisers. In GECCO'01: Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation, (pp. 892–899). San Francisco. <https://doi.org/10.5555/2955239.2955400>
- Vicsek, T., Czirók, A., Ben-Jacob, E., et al. (1995). Novel Type of Phase Transition in a System of Self-Driven Particles. *Physical Review Letters*, 75(6), 132–135. <https://doi.org/10.1103/PhysRevLett.75.1226>
- Wahby, M., Petzold, J., & Eschke, C., et al. (2019). Collective change detection: Adaptivity to dynamic swarm densities and light conditions in robot swarms. In: Artificial Life Conference Proceedings, (pp. 642–649). MIT Press: Newcastle. https://doi.org/10.1162/aisal_a_00233
- Zhang, K., Yang, Z., & Başar, T. (2021). Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handbook of Reinforcement Learning and Control* (pp. 321–384). https://doi.org/10.1007/978-3-030-60990-0_12
- Zhang, S., Liu, M. Y., Lei, X. K., et al. (2019). Dynamics and motion patterns of a k-capture game with attraction-repulsion interaction. *EPL (Europhysics Letters)*. <https://doi.org/10.1209/0295-5075/128/10003>
- Zhong, V.J., Umamaheshwarappa, R.R., & Dornberger, R., et al. (2018). Comparison of a real kilobot robot implementation with its computer simulation focussing on target-searching algorithms. In 2018 International Conference on Intelligent Autonomous Systems (ICoIAS). IEEE, (pp. 160–164). Singapore. <https://doi.org/10.1109/ICoIAS.2018.8494196>
- Zoss, B. M., Mateo, D., Kuan, Y. K., et al. (2018). Distributed system of autonomous buoys for scalable deployment and monitoring of large waterbodies. *Autonomous Robots*, 42, 1669–1689. <https://doi.org/10.1007/s10514-018-9702-0>

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.