The Impact of Agent Density and Environmental Factors on Target Tracking Swarms

Hian Lee Kwa[1], Julien Philippot[2] and Roland Bouffanais[2]

[1]Thales Solutions Asia, Singapore,
[2]University of Ottawa, Canada
hianlee.kwa@asia.thalesgroup.com

Abstract

When sizing a multi-robot swarm, a key quantity to be considered is the swarm's agent density. In the field of multi-robot and multi-agent systems, it has been acknowledged that there is a minimum agent density to ensure the emergence of cooperative behaviors, implying that too few agents within a swarm would yield an ineffective system. However, too large a swarm may result in the agents interfering with each other's actions, again resulting in subpar swarm performances. There is therefore a range of densities where swarm operations are optimal. In this study, we investigate the factors that determine this range for collective target-tracking tasks. Specifically, we show how the use of agent-based memory can reduce the density at which swarms are able to start tracking. We also show that besides strategy design, other environmental factors affect the range of densities over which swarms can operate efficaciously, such as a target's movement policy, its velocity, and the number of targets to be tracked.

Introduction

With the current ease of mass-producing automated robots and drones, many robotics researchers and practitioners alike are turning towards the use of these low-cost and easy-to-manufacture robotic units in swarming multi-robot systems (MRS). Such MRS have been shown to be capable of performing tasks more efficiently and efficaciously, and are also able to complete tasks that are simply impossible for a single agent to carry out. Given the versatility of swarming MRS, these systems have found many potential use cases, ranging from warehouse operations to space exploration (Schranz et al., 2020).

Given that the operation of an MRS involves multiple robots within a defined area, a key quantity to consider is the system's density of agents. In the field of multi-agent systems (MAS), it has long been acknowledged that there is a minimum agent density to ensure the emergence of a desired collective behavior in simplistic physics models, such as the self-propelled particles (SPP) used in Vicsek et al. (1995). Besides physics modeling, it has been shown that in both natural and man-made MAS, effective emergent collective actions require a critical minimum density of swarming agents (Khaluf et al., 2017; Schranz et al., 2021). This is because large inter-agent distances brought about by low agent densities essentially hamper the individual swarming units from cooperating with each other and carrying out the assigned mission. As a result, these low-density swarms have performances that are limited by a lack of exploitation. Indeed, it has been demonstrated by Hamann (2012) that when starting from a low density, raising the system's density by introducing more agents results in higher levels of cooperation. This is normally characterized by a superlinear increase in the system's performance.

While increasing the number of agents within a swarm may raise the system's performance due to higher levels of cooperation, the robotic units will also experience higher levels of interference, i.e., robots spend more time performing collision avoidance maneuvers (Hamann, 2013). Increases in the amounts of time and number of agents carrying out collision avoidance results in the creation of pockets of high local agent density. As more agents aggregate in these pockets, they find it harder to move away due to the increasing number of collision avoidance measures required. This results in a positive feedback loop where the high-density pockets continue to grow in size (Cates and Tailleur, 2015). This phenomenon, known as motility-induced phase separation (MIPS), exponentially increases the amount of interference experienced by the agents as the swarm density grows. As a result of this, the swarming units are hindered from moving around and exploring the environment. In turn, this reduces exploratory actions and inevitably promotes excessive levels of exploitation, which ultimately degrade the MRS's level of performance. This phenomenon is especially common in MRS tasked with clustering around points of interest in the environment, such as those performing target tracking, and would be equivalent to deploying a static sensor network or an MRS comprised of limited-mobility robots. In such cases, it may be beneficial for individual robots to withdraw from the area of interest to preserve swarm performance (Mayya et al., 2019).

The range of densities over which a swarm experiences a notable drop in performance—owing to a lack of exploration or exploitation—delimits a high-density phase and a low-density one respectively. In between these two phases sits a so-called 'transition phase', in which the performance of a swarm varies rapidly with the system's density (Hamann, 2018; Kwa et al., 2023b). It is within this transition phase that swarms are the most effective since they are able to balance the level of exploration and exploitation carried out to perform the assigned mission effectively (Kwa et al., 2022; Raoufi et al., 2023). To reduce the cost of deployed systems, swarm system designers are interested in minimizing the size of their swarms, and therefore attempt to reduce the density at which the transition phase starts. This can be achieved through algorithmic and system strategy design (Wahby et al., 2019; Kwa et al., 2023b), although it can also be influenced by other factors from the MRS's operational environment (Hecker and Moses, 2015). In contrast, operating a swarm within the high-density phase would be impractical and wasteful due to the restricted mobility caused by MIPS, as just described.

Despite growing evidence about this central range of densities in which it is practical and cost-effective to use swarming MRS, little work has been done on how the strategies deployed and various environmental factors affect this density range. As such, our contribution in this paper is a study of the various factors and their effects on the characteristics of the transition phase. This is done specifically for the case of tracking fast-moving targets, i.e., when a swarm of drones with limited sensor range is tasked with pursuing targets that can move faster than any of the individual units. The rest of the paper is structured as follows. First, we give a brief introduction to the tracking strategy used. We then show the occurrence of the transition phase regardless of the system's usage of memory and explain how the introduction of an agent-based memory can reduce the density at which the transition phase starts regardless of the target's movement type. We also explore the effect of different target parameters on when this transition phase occurs, namely a target's movement policy, its velocity, and the number of targets to be tracked simultaneously.

Methods

Tracking Strategy

The strategy used in this work was first introduced by Kwa et al. (2021). This strategy consists of two main components: (1) agent aggregation augmented with a short-term memory that facilitates exploitation of target information and, (2) inter-agent adaptive repulsion that fosters area exploration. These two components generate two velocity vectors at each time-step that, when added, yield the final agent velocity vector:

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

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 at timestep, t. Of these two components, the attractive one encourages agents to cluster around a target's location. Memory usage generates a longer-lasting point of attraction, thereby further promoting exploitative actions in situations where the target's presence may not be constantly detected. As such, each agent is given a memory that lasts for a duration of t_{mem}.

In the original paper, tuning the degree of the interconnectivity through the topological k-nearest neighbor communications network served as one method to control the system's level of exploration and exploitation carried out. Indeed, changing the degree, k, was also shown to affect a system's performance and exploration-exploitation balance regardless of density (Crosscombe and Lawry, 2021; Kwa et al., 2023b). It has also been shown previously that even when using communications networks defined by a fixed radius, a swarm's performance is still mostly dependent on the number of neighbors an agent has (Mateo et al., 2017). As such, for the results presented in this paper, k is kept constant at $k = 15$ to isolate and identify the effects of other various factors tested on the swarm's transition phase density range.

The overall strategy employed in the system is summarized in Algorithm 1. For further details regarding how $\mathbf{v}_{i,\text{att}}[t]$ and $\mathbf{v}_{i,\text{rep}}[t]$ are calculated, the reader is referred to Kwa et al. (2021).

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

Set $t = 0$, $k = 15$
while System active **do**
    for All agents **do**
        Set point of attraction
        Calculate $\mathbf{v}_{i,\text{att}}[t]$
        Calculate $\mathbf{v}_{i,\text{rep}}[t]$
        $\mathbf{v}_i[t] \leftarrow \mathbf{v}_{i,\text{att}}[t] + \mathbf{v}_{i,\text{rep}}[t]$ // Apply Eq. (1)
        $\mathbf{v}_i[t] \leftarrow (v_{\max}/v_i[t]) \cdot \mathbf{v}_i[t]$ // Ensure magnitude of velocity vector does not exceed maximum speed
        $\mathbf{x}_i[t+1] \leftarrow \mathbf{x}_i[t] + \mathbf{v}_i[t]$ // Update agent position
    end for
    $t \leftarrow t + 1$
end while

Target Representation

The considered targets move either in an evasive or non-evasive fashion. When using the non-evasive movement policy, the targets move towards random waypoints generated within the search space. While using the evasive movement policy, the targets initially follow the non-evasive policy until it makes contact with an agent. Upon contact with an agent (i.e., when an agent falls within the target's radius), the target updates its velocity according to

$$\mathbf{v}_{o,\text{rep}}[t] = -\sum_{j \in \mathcal{N}_i} \left(\frac{a_R}{r_{oj}[t]} \right)^d \frac{\mathbf{r}_{ok}[t]}{r_{oj}[t]}, \tag{2}$$

where $\mathbf{r}_{oj}[t]$ is the vector from target o to an agent j at time-step t, $\mathbf{r}_{oj}[t] = \left|\left|r_{oj}[t]\right|\right|$, and $\mathcal{N}_i$ is the set of agents considered during repulsion. In this implementation, all agents within its radius are considered to be included within this set. The overall strength of the repulsion is controlled by the constant a_R and the exponential term d in the pre-factor term (a_R/r_{oj}), set at 7.5 and 6, respectively. After encountering agents for t_{limit} consecutive iterations, the target travels in a straight line for t_{evade} time-steps in an attempt to outrun its pursuers. For the simulations conducted, $t_{\text{limit}} = 10$ and $t_{\text{evade}} = 30$.

Problem Statement

In this work, a set of N tracking agents $\mathcal{A} = \{a_1, a_2, \ldots, a_N\}$ and a set of M targets $\mathcal{O} = \{o_1, o_2, \ldots, o_M\}$, move within a bounded two-dimensional square search-space of dimensions $L \times L$ free of any obstacles. In this paper, we use the values $N = 50$, $M \in [1, 2, 3]$, and $L \in [10^{0.6}; 10^{2.65}]$. The swarm density, $\rho = N/L^2$, is varied by changing the length L of the search-space. The agents' and target's positions are denoted by $\mathbf{x}_i = (x_i, y_i)$, and maximum velocities of $\mathbf{v}_{a,\text{max}}$ and $\mathbf{v}_{o,\text{max}}$ respectively. It should be emphasized that in this work, we consider targets that move faster than their pursuing agents; as such, $\mathbf{v}_{a,\text{max}} < \mathbf{v}_{o,\text{max}}$. Furthermore, the maximum velocities are set as follows: $\mathbf{v}_{a,\text{max}} = 0.1$ and $\mathbf{v}_{o,\text{max}} \in [0.15, 0.30]$, while the agents' memory length is set such that $t_{\text{mem}} \in [0, 20]$.

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:

$$\text{cov}(o, t) = \begin{cases} 1, & \exists i \in \mathcal{A} \text{ s.t. } \|\mathbf{x}_i - \mathbf{x}_o\| \leq r, \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

where x_i is the position of an agent i and x_o is the position of a target o. Modeling the target as a binary objective function, as considered here, makes the problem akin to a visual search task where a target needs to be seen to have its presence confirmed. This is in contrast to tracking a target using the strength of a signal originating from the target (e.g., radio signal strength, chemical plume, etc.). While using the emitted signal strength permits the use of gradient-descent-based techniques, which form one of the most popular types of techniques tracking targets, such strategies become completely ineffective when a system relies on very limited sensor capabilities. The use of this type of function ensures that the system cannot use such simplistic techniques. 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 (Kwa et al., 2023b).

The goal of the system is to maximize its tracking performance (Ξ) within the environment, as measured by the following reward function:

$$\Xi = \frac{1}{TM} \sum_{t=1}^{T} \sum_{m=1}^{M} \text{cov}(o_m, t), \tag{4}$$

where T is the total time period of interest, set as $T = 100,000$. Setting T to a large value ensures the statistical stationarity of the results obtained, i.e., the system being tested is ergodic. Given that the environment's conditions stay the same, its tracking performance will tend to a stationary value as $T \to \infty$ (Khaluf et al., 2013). This means that only one simulation run is required to generate a single data point. In the simulations performed, the agents are tasked with tracking the targets in an environment free from obstacles and are assumed to be fully informed about the target's location once within its 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.

Results

Effect of Memory on the Transition Phase

From Figs. 1 and 3, it can be seen that as the swarm density increases, three separate phases, the low-density, transition, and high-density phases, can be observed based on the system's tracking performance, regardless of the usage of agent-based memory and the target's movement policy (non-evasive in Fig. 1, evasive in Fig. 3). In the low-density phase, the swarm is plainly unable to perform and does not track the target at all. The transition phase is characterized by a steep increase in a system's tracking performance as the swarm density is raised. This is followed by the high-density phase, where the target is tracked continuously, i.e., $\Xi = 1$, regardless of how it moves. Clearly, swarms cannot be used in the low-density phase due to their inability to carry out their designed tracking task. In addition, while swarms can be used to track targets in the high-density region, this task may be performed more efficiently by a static sensor network or a system of limited mobility robots. As such, swarms are best suited to conducting target tracking while operating in the transition phase. By reducing the density at which the transition phase starts, swarm designers could increase the range of conditions in which their systems can operate while also boosting the cost-effectiveness of the MRS.

Evasive Target Tracking Figure 1 suggests that an effective method to promote an earlier start of the transition phase is by implementing agent-based memory. The figure shows that swarms with memory begin their transition

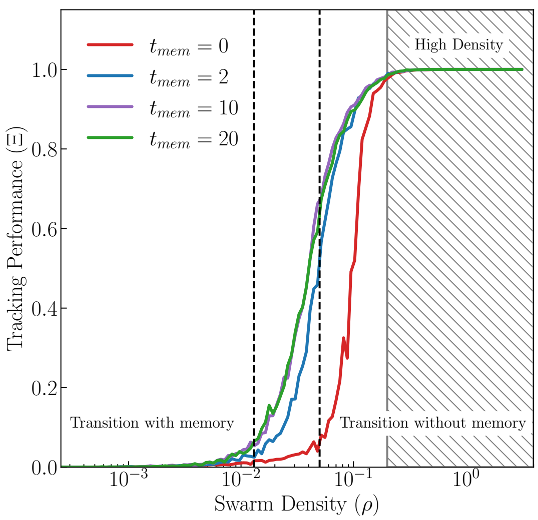

Figure 1: The effect of memory on a system's transition phase when tracking an evasive target. The start of the transition phase in a memory-less system occurs at a higher swarm density than systems with memory. The usage of memory does not affect the start of the high density phase.

phases at a lower density compared to those that do not utilize memory ($\rho \sim 1.2 \times 10^{-2}$ with memory compared to $\rho \sim 5 \times 10^{-2}$ without memory). The low-density starting occurs because memory usage introduces a persistent point of attraction when the target is found. This, in turn, allows multiple agents to gather at the target's location even though many of the swarming units may not directly observe the target, essentially promoting inter-agent cooperation and exploitative behaviors to track the target. This further emphasizes the importance of memory when tracking an evasive target as concluded in Kwa et al. (2021).

Furthermore, the swarm also appears to enter the high-density phase simultaneously regardless of the memory length used. As such, it can be said that the transition phase for the memory-less swarm is narrowed. Since a swarm is effectively unable to operate in the low-density phase and the motion of individual agents is highly restricted in the high-density phase, this suggests that there is a narrower range of conditions in which a memory-less swarming MRS can be used to carry out this target-tracking task.

While adding memory to the swarm may serve as a method to reduce the density at which the transition phase starts, it can also be seen in Fig. 1 that using excessive memory lengths does not bring further benefits to a system's tracking performance. While the starting density is greatly reduced when increasing the agents' memory length from $t_{\text{mem}} = 0$ to $t_{\text{mem}} = 2$, this reduction is not as great as the one observed when increasing the agents' memory length from $t_{\text{mem}} = 2$ to $t_{\text{mem}} = 10$. In addition, there is no difference between the transition phases of the swarms with

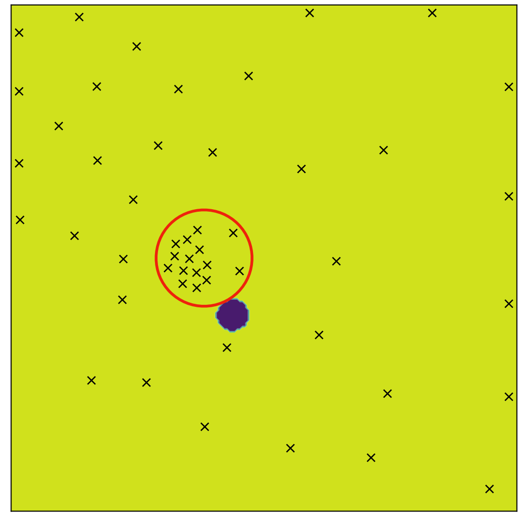

Figure 2: A swarm with a memory length of $t_{\text{mem}} = 20$ tracking an evasive target. Agents clustering in the wrong location due to the exploitation of outdated information circled in red when using excessive memory lengths.

$t_{\text{mem}} = 10$ and $t_{\text{mem}} = 20$. Indeed, as the agents' memory lengths are further increased, the agents start to exploit outdated target information and cluster at a position where the target is no longer present (see Fig. 2). As such, further augmenting the agents' memories no longer brings about any increases to the system's tracking performance despite the increase in exploitative activity.

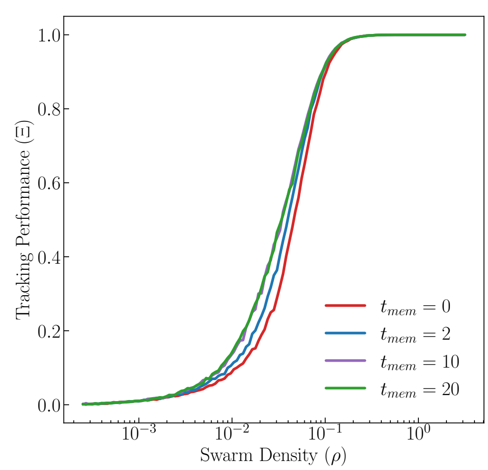

Figure 3: The effect of memory on a system's transition phase when tracking a non-evasive target.

Non-Evasive Target Tracking As seen in Fig. 3, unlike the evasive case, the absence of memory does not result hamper the system's performance at low densities. In fact, in other research works studying the tracking of non-evasive targets, the agent memory component is usually removed from the system's strategy (Hu and Eberhart, 2002; Coquet et al., 2019; Kwa et al., 2020). This is usually done as the usage of memory would encourage agents to exploit outdated information, causing individuals to cluster around a target's previous location, as seen previously with the evasive moving target. This problem is exacerbated as the target or the environment evolves at a faster rate (Jordehi, 2014). However, as can be observed in Fig. 3, using memory increases a system's performance at low densities and has virtually no effect on when it enters the high-density phase. This has therefore the positive effect of allowing the swarm to enter the transition phase at a lower density, and essentially expands the range of densities in which the use of a swarming MRS is feasible and effective.

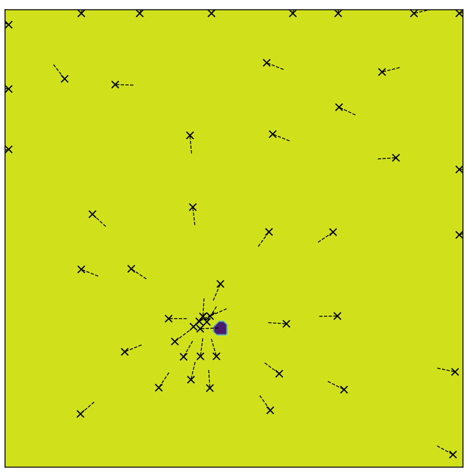

Figure 4: A swarm with a memory length of $t_{\text{mem}} = 20$ tracking a non-evasive target at a swarm density of $\rho = 10^{-2}$. Dotted lines indicate the direction of travel of the individual agents.

The reason for this increased performance at low density can be understood when analyzing Fig. 4. The introduction of memory allows agents to continue moving in the general direction of the target, even though the target may have already outrun its initial pursuers. The motion drives the agents toward the location where the target was last observed and positions them in a way that increases their likelihood of encountering the target. In addition, a large number of individuals moving in the direction of a common target also reduces the distance between agents, making it easier for them to aggregate and exploit a target's positional information once the target is found again.

Similar to the evasive target tracking case, Fig. 3 also shows that while increasing the memory length of the system may initially lead to better tracking performance, augmenting the memory even further neither results in an earlier start to the transition phase nor further improves tracking performance. Despite the benefits of implementing agent-based memory when tracking an evasive target, the figure also shows poorer levels of performance if t_{mem} is large. This is because excessive memory lengths lead to the swarming agents exploiting outdated information, causing them to aggregate for long periods of time where the target is no longer present. This flocking around a non-existent target means that the system would be unable to further improve on its tracking performance.

Effect of Target Parameters on Transition Phase

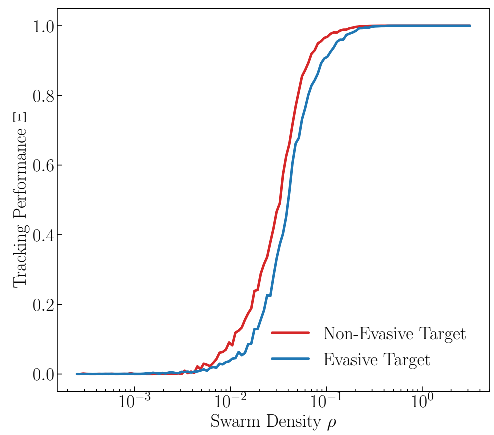

Figure 5: The effect of a target's movement policy on a tracking swarm's performance at different densities.

Target Movement Policy As previously stated, the density range over which the transition phase occurs can be influenced by factors in the operational environment, such as the parameters of the targets being tracked. One of these factors is the manner in which the target moves. As seen in Fig. 5, compared to the non-evasive target-tracking case, the transition phase occurs at a higher range of densities when the swarm is after an evasive target. This happens even when memory is used to track both types of targets and when the memory length is kept constant at $t_{\text{mem}} = 10$.

This shift in the transition phase toward a higher range of densities is to be expected as tracking an evasive target is inherently more challenging. With both the degree of the communications network topology and the memory length kept fixed, a swarm would initially show the same level of exploitation when encountering a target. However, as the target responds and tries to avoid contact with the pursuing agents, some of the exploitation carried out by the swarm is negated as the agents eventually exploit inaccurate information, thereby reducing the effective level of exploitation of the system.

In addition, after the evasive target is found, the swarm cannot follow the target as effectively as it does while tracking a non-evasive target. Due to this, the target's positional information is broadcast to fewer agents in the system and fewer agents are informed of the target's position. As such, compared to the non-evasive case, not as many agents are drawn to the evasive target, further reducing the amount of exploitation performed by the system. It is this lower level of less effective exploitation by the swarm that leads to the transition phase taking place over a higher density spectrum.

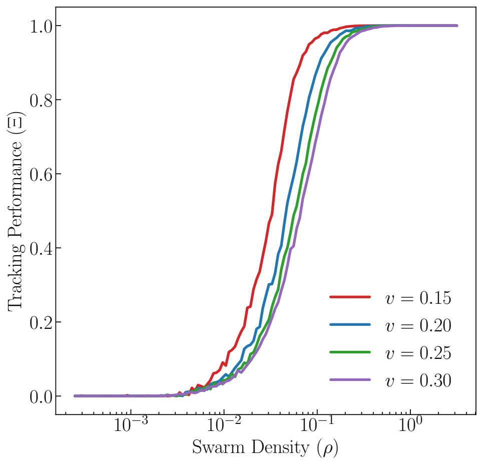

Figure 6: The effect of a non-evasive target's velocity on a tracking swarm's performance at different densities.

Target Velocity Comparable to changing a target's movement policy, increasing the target's velocity can also serve to raise the difficulty of the tracking task. Given that higher target speeds allow them to more easily outrun the pursuing swarm, some of the system's exploitative activity is negated. This occurs even though the target may be following a non-evasive movement policy. Furthermore, similar to the evasive target scenario, this also results in the target's positional information being disseminated to fewer agents, thereby reducing the amount of exploitation carried out by the swarm. As such, greater target velocities also result in the transition phase taking place over a higher range of densities due to the system's lower amount and less effective level of exploitation.

Number of Targets Unlike the two previous cases, increasing the number of targets to be tracked does not plainly increase the density range over which the transition phase occurs. Instead, it can be seen in Fig. 7 that the transition phase starts at a lower density when more targets are to be tracked. As the density of the swarm increases, systems tracking fewer targets start to outperform those tracking greater numbers of targets. Eventually, swarms that track fewer targets enter the high-density phase earlier than those tracking many targets simultaneously.

This observation can easily be understood because the likelihood of encountering and pursuing a single target increases as the number of targets present in the environment grows. At this point, it is worth remembering that at low density, the tracking performance of swarms tends to be limited by a lack of exploitation. The higher likelihood of encountering a target raises the level of exploitative activity performed by the swarm carrying out multi-target tracking, causing it to enter the transition phase earlier, i.e., at a lower density.

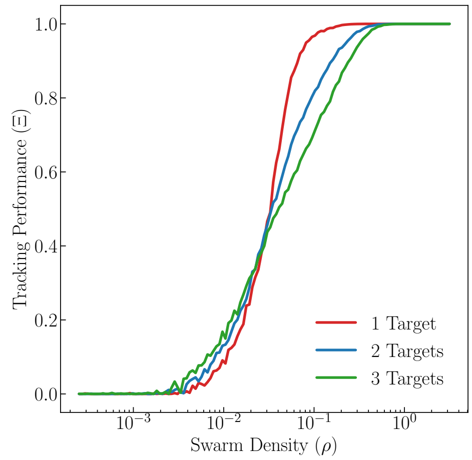

Figure 7: The effect of the number of non-evasive target's to be tracked simultaneously on a tracking swarm's performance at different densities.

On the other hand, when operating at high density, swarm performance tends to be limited by a lack of exploration. As such, the elevated level of exploitation encouraged by the increased number of target encounters results in even less exploration being carried out by the system. Combined with the increased demand for exploration in the pursuit of multiple targets, having agents being drawn away from exploring the environment results in a relatively inferior tracking performance at high density compared to single-target tracking. Consequently, this leads to a later start to the high-density phase, where the targets are tracked continuously.

Discussion

In the design of an MRS swarm, system designers are pushed towards developing smaller swarms in an attempt to reduce the system's overall cost—production, maintenance, and potentially upgrade. However, as seen in the results presented in this paper, using too few agents in a large environment results in an ineffective swarm that cannot perform the desired task. Conversely, while systems can be used in high swarm density environments, using a static sensor array may be more efficient. As such, swarming MRS are well suited to operate within the transition phase, where small changes in the swarm density yield large changes to the system's performance.

To increase the viability of the usage of swarming MRS, the density at which the transition phase begins needs to be reduced, thereby allowing these systems to operate over a broader and more challenging range of conditions. To enable this, the level of exploitation carried out at low swarm density must be increased. One possible method of achieving this is by introducing an agent-based memory to the swarm's agents. Doing so generates a longer-lasting point of attraction, thereby promoting higher levels of exploitative activity and lowering the density at which the transition phase begins. This can be done regardless of whether the target is following an evasive or non-evasive movement policy, although increasing memory length past a certain point does not further improve the system's tracking performance.

In addition, we have also shown that a system's transition phase can be affected not only by the swarm strategy used but also by its operational environment—specifically the characteristics of the target being pursued. When the target is more challenging to follow, either due to its increased speed or its movement policy, the range of densities over which the transition phase occurs increases and may even narrow. This is because the target's movements negate the swarm's exploitative actions. In addition, agents in the swarm disseminate the target's positional information to fewer neighbors, ultimately reducing the amount of exploitation carried out by the entire system.

Not all changes to the environment plainly increase or decrease the transition phase's density range occurs. When tracking multiple targets, the transition phase started earlier due to the increased number of target encounters increasing the exploitation level. However, this high level of exploitation also delayed the start of the high-density phase as the near-constant tracking of a greater number of targets also requires elevated levels of exploration.

A limitation of this work was that the simulations were carried out in an environment free of obstacles, as well as with ideal communications between agents. The simulations did not include factors such as packet dropouts, noise, time delays, and bandwidth limitations that could hamper inter-agent transmissions. The distance between communication neighbor pairs and the communications range of individual agents were also not considered. While this does not accurately reflect the conditions an MRS would experience in a real environment, this work focuses on the highly dynamic problem of tracking a fast-moving target and the effect of operating swarming MRS in different conditions at various swarm densities. While using imperfect communications, the amount of exploitation carried out by a system would clearly be hampered further; there would be a smaller subset of neighbors with which an agent can communicate and the information transmitted may be inaccurate due to noise or time delays in communication. This is especially true in low agent density environments where adding one more neighbor may require a substantial increase in an agent's communications radius. Similarly, when operating in an environment cluttered with obstacles, agents would not be able to take the most direct path to the target, reducing the effectiveness of an agent's exploitative actions. While the effects of these more realistic scenarios may be apparent. In doing so, system designers would be able to identify and prioritize the factors that need to be considered when designing swarm strategies.

Besides more realistic environmental conditions, studies should also be performed on swarms carrying out other types of missions. This work was also only carried out in the context of target tracking and the results may not generalize when swarms are implemented in other tasks. For example, we concluded that bringing forward the high-density phase, i.e., making it occur at a lower density, would be beneficial for the system as mobile swarming robots can be replaced by a static sensor network or by robots with a lower level of mobility to save on unit production costs without sacrificing tracking performance. However, in cases such as resource foraging or construction, where the use of static agents is not feasible, these high-density swarms would ultimately impede agent movement to the detriment of the system due to the high level of interference in the swarm and MIPS (Rosenfeld et al., 2006).

While the work in this paper does not present new methods that can be utilized by swarming MRS, it offers several insights into how changes to the swarm's strategy and environment can change a system's performance at different swarm densities. As is evident in the presented results, efforts into developing more advanced swarm strategies should focus on reducing the density at which the transition phase starts, thereby allowing swarms to operate at lower densities and lowering the production costs of such systems. Such advances to swarm algorithms may come from the field of multi-agent reinforcement learning (MARL) that could potentially allow a system to take advantage of agents taking on different roles while performing the task or better allow an agent to consider its neighbors' actions and local observations in its decision making process (Kouzehgar et al., 2020, 2023). In doing so, swarms can be designed to be more adaptive and change their actions and behaviors to suit their operating environment as it evolves, thereby yielding higher performing MRS with higher levels of swarm intelligence (Kwa et al., 2023a).

Acknowledgements

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

References

Cates, M. E. and Tailleur, J. (2015). Motility-induced phase separation. *Annual Review of Condensed Matter Physics*, 6(1):219–244.

Coquet, C., Aubry, C., Arnold, A., and Bouvet, P.-J. (2019). A local charged particle swarm optimization to track an underwater mobile source. In *OCEANS 2019 - Marseille*, Marseille, France. IEEE.

Crosscombe, M. and Lawry, J. (2021). The Impact of Network Connectivity on Collective Learning. In *Proceedings of the 15th International Symposium on Distributed Autonomous Robotics Systems (DARS21)*.

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.*, pages 168–179, Berlin, Germany. Springer.

Hamann, H. (2013). Towards swarm calculus: Urn models of collective decisions and universal properties of swarm performance. *Swarm Intelligence*, 7(2-3):145–172.

Hamann, H. (2018). 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 LNCS:31–42.

Hecker, J. P. and Moses, M. E. (2015). Beyond pheromones: evolving error-tolerant, flexible, and scalable ant-inspired robot swarms. *Swarm Intelligence*, 9(1):43–70.

Hu, X. and Eberhart, R. C. (2002). Adaptive particle swarm optimization: Detection and response to dynamic systems. In *Proceedings of the 2002 Congress on Evolutionary Computation, CEC 2002*, volume 2, pages 1666–1670, Honolulu, HI, USA.

Jordehi, A. R. (2014). Particle swarm optimisation for dynamic optimisation problems: A review. *Neural Computing and Applications*, 25:1507–1516.

Khaluf, Y., Birattari, M., and Rammig, F. (2013). Probabilistic analysis of long-term swarm performance under spatial interferences. In *International Conference on Theory and Practice of Natural Computing*, pages 121–132, Caceres, Spain.

Khaluf, Y., Pinciroli, C., Valentini, G., and Hamann, H. (2017). The impact of agent density on scalability in collective systems: Noise-induced versus majority-based bistability. *Swarm Intelligence*, 11(2):155–179.

Kouzehgar, M., Meghjani, M., and Bouffanais, R. (2020). Multi-agent reinforcement learning for dynamic ocean monitoring by a swarm of buoys. In *Global OCEANS 2020: Singapore-U.S Gulf Coast*, Singapore.

Kouzehgar, M., Song, Y., Meghjani, M., and Bouffanais, R. (2023). Multi-target pursuit by a decentralized heterogeneous UAV swarm using deep multi-agent reinforcement learning. In *2023 International Conference on Robotics and Automation (ICRA)*, London, UK. IEEE.

Kwa, H. L., Kit, J. L., and Bouffanais, R. (2020). Optimal Swarm Strategy for Dynamic Target Search and Tracking. In *Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020)*, pages 672–680, Auckland, New Zealand.

Kwa, H. L., Kit, J. L., and 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.

Kwa, H. L., Kit, J. L., and Bouffanais, R. (2022). Balancing collective exploration and exploitation in multi-agent and multi-robot systems: A review. *Frontiers in Robotics and AI*, 8(771520).

Kwa, H. L., Kit, J. L., Horsevad, N., Philippot, J., Savari, M., and Bouffanais, R. (2023a). Adaptivity: A path towards general swarm intelligence? *Frontiers in Robotics and AI*, 10(1163185).

Kwa, H. L., Philippot, J., and Bouffanais, R. (2023b). Effect of swarm density on collective tracking performance. *Swarm Intelligence*.

Mateo, D., Kuan, Y. K., and Bouffanais, R. (2017). Effect of correlations in swarms on collective response. *Scientific Reports*, 7(1):1–11.

Mayya, S., Pierpaoli, P., and Egerstedt, M. (2019). Voluntary retreat for decentralized interference reduction in robot swarms. In *Proceedings - IEEE International Conference on Robotics and Automation*, pages 9667–9673, Montreal, Quebec, Canada.

Raoufi, M., Romanczuk, P., and Hamann, H. (2023). Estimation of continuous environments by robot swarms: Correlated networks and decision-making. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, London, UK.

Rosenfeld, A., Kaminka, G. A., and Kraus, S. (2006). A study of scalability properties in robotic teams. In Scerri, P., Vincent, R., and Mailler, R., editors, *Coordination of Large-Scale Multiagent Systems*, pages 27–51. Springer, Boston, MA, USA.

Schranz, M., Di Caro, G. A., Schmickl, T., Elmenreich, W., Arvin, F., Şekercioğlu, A., and Sende, M. (2021). Swarm intelligence and cyber-physical systems: Concepts, challenges and future trends. *Swarm and Evolutionary Computation*, 60(100762).

Schranz, M., Umlauft, M., Sende, M., and Elmenreich, W. (2020). Swarm robotic behaviors and current applications. *Frontiers in Robotics and AI*, 7(36).

Vicsek, T., Czirók, A., Ben-Jacob, E., Cohen, I., and Shochet, O. (1995). Novel type of phase transition in a system of self-driven particles. *Physical Review Letters*, 75(6):132–135.

Wahby, M., Petzold, J., Eschke, C., Schmickl, T., and Hamann, H. (2019). Collective change detection: Adaptivity to dynamic swarm densities and light conditions in robot swarms. In *Artificial Life Conference Proceedings*, pages 642–649, Newcastle, United Kingdom. MIT Press.