



The Exploration and Exploitation Dynamics of Target Tracking Swarms

Submitted by

Hian Lee KWA

Thesis Advisor

Assoc. Prof. Roland Bouffanais

Administrative Thesis Advisor

Assoc. Prof. Yuen Chau

Engineering Product Development Pillar

A thesis submitted to the Singapore University of Technology and Design in
fulfillment of the requirement for the degree of Doctor of Philosophy

2022

PhD Thesis Examination Committee

Assoc. Prof. Foong Shaohui (Engineering Product Design, SUTD), TEC Chair

Assoc. Prof. Yuen Chau (Engineering Product Design, SUTD), Administrative Thesis Advisor

Assoc. Prof. Roland Bouffanaïs (Mechanical Engineering, University of Ottawa), Thesis Advisor

Assoc. Prof. Tan U-Xuan (Engineering Product Design, SUTD), Internal TEC Member

Asst. Prof. Malika Meghjani (Computer Science and Design, SUTD), Internal TEC Member

Abstract

Engineering Product Development Pillar

Doctor of Philosophy

The Exploration and Exploitation Dynamics of Target Tracking Swarms

by Hian Lee KWA

Can a multi-robot system (MRS) be made to find and track a target that can move faster than any of its component robots? Such tasks have long been considered impossible due to the assumption that the target will always be able to outrun the individual robots. The work done in this thesis shows that this task is in fact, achievable and at its root, boils down to the exploration-exploitation dilemma—the choice a system must make between gathering more information about the environment or making use of the information currently available.

To accomplish the task of tracking a fast-moving non-evasive target, a fully decentralised search and track strategy based on the Particle Swarm Optimisation (PSO) algorithm is used. This strategy is complemented by an adaptive inter-agent repulsion behaviour, used to promote exploration, as well as an adjustable k -nearest neighbour communications network, used to tune the system's exploration-exploitation balance. To achieve the more challenging task of tracking an evasive target, the individual agents of the swarming MRS are endowed with a short-term memory, thereby promoting higher levels of exploitation. The two developed strategies are then validated through physical tests using a decentralised swarm of miniature ground robots.

Through both virtual and physical experimentation, an optimum level of connectivity to maximise the MRS's tracking performance is revealed. The origin of this optimum level of connectivity is further traced back to an optimum balance in the amount of exploratory and exploitative actions carried out by the system. The effect of various environmental factors and mission parameters on this optimum, such as the swarm density, number of agents used, and the movement profile of the targets, are also studied. The results presented in this thesis further emphasises the importance of attaining the correct exploration-exploitation balance when developing a swarm strategy. This optimum changes according to the task set for the system and the results presented shed some light on how to tune a swarm's exploration-exploitation dynamics to find this optimum, potentially paving the way for better swarming algorithms to be developed.

Publications

- Kwa, H. L., Philippot, J., & Bouffanais, R. (*In Review*) "Effect of Swarm Density on Collective Tracking Performance". Swarm Intelligence.
- Kwa, H. L., Babineau, V., Philippot, J., & Bouffanais, R. (*In Press*) "Adapting the Exploration-Exploitation Balance in Heterogeneous Swarms: Tracking Evasive Targets". Artificial Life. (Kwa et al., 2022)
<https://arxiv.org/abs/2207.13523>
- Kwa, H. L., & Bouffanais, R. (2022). "The Effect of Network Connectivity on Exploration and Exploitation During Decentralized Collective Learning". 2022 International Workshop on Agent-Based Modelling of Human Behaviour (ABMHuB). (Kwa and Bouffanais, 2022)
http://abmhub.cs.ucl.ac.uk/2022/camera_ready/Kwa_Bouffanais.pdf
- Horsevad, N., Kwa, H. L., & Bouffanais, R. (2022). "Beyond Bio-Inspired Robotics: How Multi-Robot Systems Can Support Research on Collective Animal Behavior". Frontiers in Robotics and AI, 9(865414). (Horsevad, Kwa, and Bouffanais, 2022)
<https://doi.org/10.3389/frobt.2022.865414>
- 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, 8(771520). (Kwa, Kit, and Bouffanais, 2022)
<https://doi.org/10.3389/frobt.2021.771520>
- Kwa, H. L., Kit, J. L., & Bouffanais, R. (2021). "Tailoring Exploration and Exploitation in Multi-Agent Systems with Short-Term Memory and Limited Social Interaction". 2021 International Workshop on Agent-Based Modelling of Human Behaviour (ABMHuB). (Kwa, Kit, and Bouffanais, 2021a)
http://abmhub.cs.ucl.ac.uk/2021/camera_ready/Kwa_et_al.pdf
- Kwa, H. L., Kit, J. L., & Bouffanais, R. (2021). "Tracking Multiple Fast Targets With Swarms : Interplay Between Social Interaction and Agent Memory". ALIFE 2021: The 2021 Conference on Artificial Life. (Kwa, Kit, and Bouffanais, 2021b)
https://doi.org/10.1162/isal_a_00376
- Kwa, H. L., Tokić, G., Bouffanais, R., & Yue, D. K. P. (2020). "Heterogeneous Swarms for Maritime Dynamic Target Search and Tracking". Global OCEANS 2020: Singapore-U.S Gulf Coast. (Kwa et al., 2020)
<https://doi.org/10.1109/IEEECONF38699.2020.9389145>
- Kwa, H. L., Kit, J. L., & Bouffanais, R. (2020). "Optimal Swarm Strategy for Dynamic Target Search and Tracking". Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), 672–680. (Kwa, Kit, and Bouffanais, 2020)
<https://doi.org/10.5555/3398761.3398842>

Acknowledgements

One does not go through a PhD programme alone. The journey that I have been on for the last four years has been no different. During this time, many have given their support and without them, I would not have had such an enjoyable time doing this programme, or even be at the point of submitting and defending my thesis. There are some people whom I acknowledge here, but there are many I do not mention to whom I also owe my gratitude.

First and foremost, to my advisor, Prof. Roland Bouffanais. Probably the most common response I get when I tell my friends what you let me do or the things that you do for me is, "*wahlao*, how come your professor so good one?" You were constantly supporting and guiding me as I went about figuring out how to go about my research and gave me the space to try out new things and pursue other research interests. Not to mention the many late nights you spent with me going through results and manuscripts. I am a better researcher for having been your PhD student.

To Prof. Yuen Chau, thank you for agreeing and coming on as my internal thesis advisor for the last year and a half of my PhD programme and all the support you have given me since.

To my industrial supervisors from Thales, Hervé Jarry, Tan Dunlin, Gan Yung Sze, Chai Teck Yoong & Ng Chea Siang for accepting me onto the IPP programme in the first place, then for supporting my research and helping me to keep it grounded in the industrial applications, and giving me inspiration and ideas to try whenever I got stuck. To the other Thales IPPs: Hong Jie, Marvin, Chai Yan, Wen Hao, Kah How, Zhi Lin, Samson & Abigail, thank you for your friendship, camaraderie, and guiding the most junior of the IPPs throughout the whole PhD process.

To those who have helped with my research and collaborated with me over the last few years, Jabez, Nikolaj, Hari, Victor, Julien & Michael. It was a pleasure working with all of you and discussing our findings together. Also, please know that you all have saved me many headaches from trying to make the robots work and running and doing the analyses for 1001 more simulations by myself. For that I am extremely grateful.

To my many friends who supported me from the sidelines or did their PhDs alongside mine, thank you for being there, celebrating my small successes, sharing in my frustrations, and all the memes in between. Thank you for listening to me going on about my research (even though I might not have made much sense to you), as well as to my complaints about the reviews that I received and about the papers that I reviewed. Thank you for the surf sessions, the rolls, and ~~tapping~~ to tolerating my wrist locks. Probably most importantly, thank you for the food, the many KBBQ dinners, or even just coming to the cafe to chill. All of these made my PhD journey that little bit easier.

Finally, to my parents, my favourite (only) sister & Gen, words cannot express how much I owe to all of you to be at this point. For the love and support you have given me, for the sanity checks, for all the times you have told me to just do it, for everything. Thank you. You have been there through all of my good and bad moments. Me being at this point in my life most certainly would not have been possible without you.

Contents

PhD Thesis Examination Committee	i
Abstract	ii
Publications	iii
Acknowledgements	iv
1 Introduction	1
1.1 Swarm Intelligence Algorithms	2
1.2 Swarm Robotics	2
1.3 Swarm Communications	4
1.4 Target Search and Tracking	4
1.5 Research Objectives and Approach	5
1.6 Contributions	5
1.7 Thesis Outline	6
2 Balancing Exploration and Exploitation	8
2.1 Introduction	8
2.2 Characterising Exploration and Exploitation	9
2.3 Measuring the Exploration-Exploitation Balance	10
2.3.1 Spatial Distribution Based Metrics	11
2.3.2 Probability Based Metrics	12
2.3.3 Other Metrics	14
2.4 Agent Response Methods	15
2.4.1 Changes in State	16
Self-Determined State Changes	16
Assigned State Changes	20
2.4.2 Small Response Changes	23
Attraction-Repulsion Dynamics	23
Random Walks	26
Time Intervals	27
2.5 Information Dissemination Methods	28
2.5.1 Changes in Network Topology	30
Number of Neighbours	30
Communication Range	31
Forced Network Changes	31
2.5.2 Stigmergy	32
2.6 Summary	33

3	Non-Evasive Targets	35
3.1	Introduction	35
3.2	Related Work	36
3.2.1	Particle Swarm Optimisation	36
3.2.2	Network Connectivity	37
3.2.3	Heterogeneous Agents	38
3.3	Search and Track Strategy	39
3.3.1	Swarm Connectivity	40
3.3.2	Attraction and Aggregation	40
3.3.3	Adaptive Repulsion	41
3.4	Simulation Methods	42
3.4.1	Target Representation	42
3.4.2	Problem Statement	43
3.4.3	Swarm Performance Metrics	43
	Tracking Performance	43
	Exploration and Exploitation Dynamics	44
3.4.4	Simulation Setup	46
3.5	Simulation Results	46
3.5.1	Homogeneous Swarms	46
	Cumulative Velocity Fluctuation Magnitude	46
	Heading-Bearing Correlation	47
	Engagement Ratio	49
	Tracking Performance	51
3.5.2	Strategy Comparisons	54
3.5.3	Heterogeneous Swarms	56
	Tracking Performance	56
	Cumulative Velocity Fluctuation Magnitude	56
	Heading-Bearing Correlation	58
3.6	Robotic Experiments	59
3.6.1	Swarm Robotic System	59
3.6.2	Experimental Conditions	61
3.6.3	Experimental Results	62
	Qualitative Observations	62
	Tracking Performance	62
	System Response	62
3.7	Discussion	63
4	Evasive Targets	65
4.1	Introduction	65
4.2	Related Work	66
4.2.1	Chase and Capture Patterns	66
4.2.2	Agent-Based Memory	67
4.3	Memory-Augmented Search and Track Strategy	68
4.4	Simulation Methods	70
4.5	Simulation Results	70
4.5.1	Homogeneous Swarms	70
	Impact of Degree of Network Connectivity	70

	Impact of Agent Memory Length	73
	Balancing Network Connectivity and Memory	74
4.5.2	Strategy Comparisons	76
4.5.3	Heterogeneous Swarms	77
	Tracking Performance	77
	Exploration-Exploitation Dynamics	78
4.5.4	Heterogeneous Strategies	80
4.6	Robotic Experiments	83
4.6.1	Experiment Conditions	83
4.6.2	Experimental Results	83
4.7	Discussion	84
5	Swarm Density	86
5.1	Introduction	86
5.2	Related Work	88
5.2.1	Swarm Size and Exploration	88
5.2.2	Marginal Utility	89
5.2.3	Motility-Induced Phase Separation	90
5.3	Simulation Methods and Metrics	90
5.4	Results	91
5.4.1	Effect of Swarm Density on Tracking Performance	91
5.4.2	Effect of Swarm Density on Exploration-Exploitation Dynamics	92
5.4.3	Effect of Swarm Size	97
5.5	Discussion	100
6	Conclusions and Future Outlook	102
6.1	Summary of Findings	102
6.2	Discussion and Outlook	104
A	Local Density	109
B	Factors Affecting the Transition Phase	111
C	Result Variability	114
	Bibliography	118

List of Figures

2.1	Classification of different agent response methods to control an MAS' exploration-exploitation balance.	16
2.2	Flowchart detailing how an agent deterministically decides on its response when provoked by environmental stimuli.	17
2.3	Inter-agent interaction forces generated by various repulsion and attraction schemes. Positive values indicate an attractive force and negative values indicate a repulsive force.	25
2.4	Classification of different information dissemination methods to control an MAS' exploration-exploitation balance.	28
3.1	Bearing of the target ($T[t]$) relative to the agent ($t_i[t]$) and the heading of the agent ($v_i[t]$) at time t	45
3.2	A system's cumulative velocity fluctuation magnitude, Γ , with varying levels of connectivity, k , and at two different target speeds, v , while using adaptive repulsion (solid line) and constant repulsion (dashed line). . . .	47
3.3	Distribution of the heading-bearing correlations, $\phi_i[t] = \cos \theta_i[t]$, (x -axis) for different levels of the swarm connectivity, k , target speeds, v . "Adaptive Repulsion" corresponds to a swarm utilising the newly proposed repulsion scheme, while "Constant Repulsion" considers the minimal static repulsion at $a_{R,\min}$	48
3.4	(Left) Screenshot of a simulated system using the adaptive repulsion scheme. Agents in the upper portion of the environment are being blocked by the search-space boundaries and neighboring agents. (Right) Screenshot of a simulated system using the constant repulsion scheme. All agents remaining in place after losing track of the target with the low exploratory behaviour in the presence of low constant repulsion.	49
3.5	Swarm system engagement ratio, Θ , while tracking a single non-evasive target traveling at different speeds. Maximum agent velocity set at $v_{a,\max} = 0.1$	50
3.6	Swarm system engagement ratio, Θ , while tracking multiple non-evasive targets traveling at different speeds. Maximum agent velocity set at $v_{a,\max} = 0.1$. Maximum Target velocity set at $v_{o,\max} = 0.2$	51
3.7	Tracking performance of a swarm consisting of 50 agents pursuing a single non-evasive target travelling at different velocities. Swarming agents travelled at a maximum speed of $v_{a,\max} = 0.1$. The solid line represents a swarm utilising the adaptive repulsion scheme with a variable a_R , while the dotted line represents a swarm utilising a constant repulsion scheme with a constant a_R	52

3.8	Tracking performance of a swarm pursuing multiple non-evasive targets with a speed of $v_{o,\max} = 0.2$. Pursuing agents travelled at a maximum speed of $v_{a,\max} = 0.1$	53
3.9	Distribution of agents (black crosses) while tracking multiple targets (blue circles). Agents using a lower level of connectivity more readily form multiple clusters, circled in red, to track targets (left). Agents with higher levels of connectivity tend to form one large cluster that can only track a single target (right).	54
3.10	Tracking performance of swarms using the proposed search and track strategy with (1) a $k = 0$ network, (2) a $k = 15$ network, and (3) a $k = 49$ network, as well as a swarm using the standard PSO algorithm. The target moved at a maximum speed of $v_{o,\max} = 0.2$ while the pursuing agents travelled at a maximum speed of $v_{a,\max} = 0.1$	55
3.11	Tracking performance of a heterogeneous swarm pursuing a non-evasive target moving at a speed of $v_{o,\max} = 0.3$. Slow agents move at a speed of $v_{s,\max} = 0.1$ and fast agents move at a speed of $v_{f,\max} = 0.26$. (A): A heterogeneous system with a level of connectivity, k , is maintained at $k = 20$ while the proportion of fast agents in the swarm is varied. (B): A heterogeneous system where the agent composition of 15 fast agents and 35 slow agents compared to a homogeneous system comprised of 50 slow agents. The performance of the two systems are compared while k is varied.	55
3.12	The cumulative velocity fluctuation magnitude, Γ , of a swarm with varying proportions of fast agents using a network degree of $k = 20$. 50 agents were simulated tracking a single non-evasive target travelling at a speed of $v_{o,\max} = 0.3$. Slow agents moved at a speed of $v_{s,\max} = 0.1$ and fast agents moved at a speed of $v_{f,\max} = 0.26$	57
3.13	The cumulative velocity fluctuation magnitude, Γ , of a swarm of 50 agents comprised of 15 fast agents with varying levels of connectivity, k . The system tracked a single non-evasive target travelling at a speed of $v_{o,\max} = 0.3$. Slow agents moved at a speed of $v_{s,\max} = 0.1$ and fast agents moved at a speed of $v_{f,\max} = 0.26$	57
3.14	Distribution of heading-bearing correlations, $\phi_i[t]$, for different levels of the swarm connectivity, k , and target speeds, v . The upper row shows the heading-bearing correlations for a heterogeneous swarm comprised of 15 fast agents and 35 slow agents, while the lower row shows the heading-bearing correlations for a homogeneous swarm of 50 slow agents.	58
3.15	A heterogeneous swarm comprised of 35 slow agents (black crosses) and 15 fast agents (blue crosses) tracking a disc-shaped target (dark blue). Agents to the left of the target are locked in position due to them attaining their maximum repulsion strength and lack of target information.	59
3.16	Orion robotic platform used in the swarm robotic experiments (Kit et al., 2019).	60

3.17	(Top) Positions of the target, represented by the red circle, the robots, and path reconstruction of the robots as observed through a central monitoring system, represented by the coloured dots and their associated trails. (Bottom) Photos from the physical experiment using a connectivity of $k = 6$ and adaptive repulsion. Each column corresponds to a different instant during the collective search and tracking run.	61
4.1	Common chase strategies used by agents (blue) while pursuing an evasive target (red). (A) Linear target chase, (B) Encirclement of target, and (C) Driving of target into other agents. (Figure adapted from Kamimura and Ohira (2019).)	66
4.2	Non-evasive target (solid line) and evasive target (dashed line) tracking performance with different maximum target speeds, $v_{o,\max}$. Maximum agent's velocity set at $v_{a,\max} = 0.1$ and memory length of $t_{\text{mem}} = 20$ was given to the agents.	71
4.3	Swarm system engagement, Θ , with a single evasive target traveling at different maximum speeds, $v_{o,\max}$. The maximum agent's velocity was set at $v_{a,\max} = 0.1$ and memory length was set at $t_{\text{mem}} = 20$	72
4.4	Multiple non-evasive (solid lines) and evasive (dashed lines) target tracking performance (maximum agent's velocity $v_{a,\max} = 0.1$, memory length of $t_{\text{mem}} = 20$, and maximum targets' velocity $v_{o,\max} = 0.2$).	73
4.5	Single evasive and non-evasive target tracking performance with different implemented agent-based memory lengths (target speed $v_{o,\max} = 0.2$). Tracking agents are connected in a dynamic $k = 20$ network.	74
4.6	Evasive target system engagement with varying memory lengths. Target moved at a speed of $v_{o,\max} = 0.2$. Agents were connected in a dynamic $k = 20$ network.	75
4.7	Single non-evasive (solid line) and an evasive (dashed line) target tracking performance while varying the length of the implemented agent-based memory and degree of network connectivity.	75
4.8	Engagement-Tracking plots of a swarm with different t_{mem} and k tracking an evasive (left) and a non-evasive (right) target. Darker shaded points indicate swarms using networks with higher values of k	76
4.9	Tracking performance of swarms using the proposed search and track strategy with (1) a $k = 0$ network with memory, (2) a memory-less $k = 10$ network, (3) a $k = 10$ network with memory, and (4) a $k = 49$ network with memory, as well as a swarm using the standard PSO algorithm. The target moved at a maximum speed of $v_{o,\max} = 0.2$ while the pursuing agents travelled at a maximum speed of $v_{a,\max} = 0.1$	77
4.10	Tracking performance of a heterogeneous swarm tracking evasive multiple targets using a memory length of $t_{\text{mem}} = 20$. Simulations for (A) were performed using a network connectivity of $k = 12$ for different proportions of fast agents. Simulations for (B) were performed using 15 fast agents and 35 slow agents for different levels of connectivity.	78

4.11	Heterogeneous swarm comprised of 35 slow-moving agents (black crosses) and 15 fast-moving agents (blue crosses) tracking an evasive target (dark blue disc). (Left) Fast agents clustering while pursuing a target; (Right) After losing track of the target, fast agents distribute themselves around the system due to the variable inter-agent repulsion.	79
4.12	Swarms comprised of 50 agents with different agent compositions of fast and slow tracking a single fast-moving evasive target. (A) Tracking performance of the system while operating with different levels of connectivity k with the optimum k decreasing as the number of fast agents present increases. (B) Engagement-Tracking plots where darker shaded points indicate swarms communicating through networks with higher values of k	79
4.13	Engagement ratio of a heterogeneous swarm tracking evasive multiple targets using a memory length of $t_{\text{mem}} = 20$. Simulations for (A) were performed using a network connectivity of $k = 12$ for different proportions of fast agents. Simulations for (B) were performed using 15 fast agents and 35 slow agents for different levels of connectivity.	80
4.14	Evasive tracking performance of homogeneous swarms with different $v_{a,\text{max}}$, and memory lengths, t_{mem} . Simulations were run at $k = 14$ and $v_{o,\text{max}} = 0.3$	81
4.15	Single evasive target tracking performance of a heterogeneous comprised of 30 slow agents and 10 fast agents with slow and fast agents, each having different levels of connectivity.	82
4.16	(Top) Reconstruction of target (black circle) and robot positions and robot paths (coloured traces) as observed through our monitoring system. (Bottom) Physical experiments using a connectivity of $k = 5$ and adaptive repulsion. (A) Robots initialised at the start of the experiments. (B) Robots without agent-based memory unable to respond to a target's presence. (C) Robots with agent-based memory moving towards target after being found.	82
5.1	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 $v_{o,\text{max}} = 0.15$. 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	92
5.2	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 $v_{o,\text{max}} = 0.15$	93
5.3	Tracking performance, Ξ , and exploration ratio, Λ , of a swarm of $N = 50$ agents using various levels of connectivity, k , while tracking a non-evasive target traveling at $v_{o,\text{max}} = 0.15$	94

5.4	A swarm of $N = 50$ agents tracking a fast-moving non-evasive target traveling at $v_{o,\max} = 0.15$. 5.4a: 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 maximises Ξ , is highlighted in grey. 5.4b: Relationship between a swarm's exploration ratio (Λ) and the average swarm density while operating at different levels of connectivity. 5.4c: 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, Λ^* , 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 (see Fig. 5.1).	96
5.5	Exploration ratios 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 $v_{o,\max} = 0.15$.	98
5.6	Exploration ratios 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 $v_{o,\max} = 0.15$.	99
5.7	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).	100
5.8	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 $v_{o,\max} = 0.15$. Tracking performances are plot against k (left) and k/N (right).	101
6.1	Comparison the search and track strategy presented in Chapter 4 and a rudimentary memory-less strategy that uses a constant inter-agent repulsion behaviour. 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.	107
A.1	(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).	110

B.1	The tracking performance and engagement ratio of swarm of 50 agents tracking different types of targets. (A) and (B): Tracking a single evasive and non-evasive target with $v_{o,\max} = 0.15$ and level of connectivity set at $k = 15$. (C) and (D): Tracking targets with varying $v_{o,\max}$ that follow a non-evasive movement policy. with the level of connectivity set at $k = 15$. (E) and (F): Single target tracking (solid line), triple target tracking (dotted line) with various levels of connectivity. The target travels at a maximum speed of $v_{o,\max} = 0.15$ following a non-evasive movement policy.	112
C.1	Companion figure to Fig. 3.5 with uncertainty ranges included.	115
C.2	Companion figure to Fig. 3.7 with uncertainty ranges included.	115
C.3	Companion figure to Fig. 3.8 with uncertainty ranges included.	116
C.4	Companion figure to Fig. 4.5 with uncertainty ranges included.	116
C.5	Companion figure to Fig. 4.3 with uncertainty ranges included.	117
C.6	Companion figure to Fig. 4.4 with uncertainty ranges included.	117

List of Tables

2.1	List of works using spatial distribution metrics to quantify MAS exploration and exploitation.	13
2.2	List of works using probability based metrics to quantify MAS exploration exploitation dynamics.	14
2.3	List of works using self-determined state change strategies to influence a system's exploration and exploitation dynamics.	18
2.4	List of works using assigned agent responses to influence a system's exploration and exploitation dynamics.	21
2.5	List of works using small response changes to influence the level of exploration and exploitation of an MAS.	24
2.6	List of works using different information dissemination strategies to influence the level of exploration and exploitation of an MAS or MRS. . . .	29
2.7	List of works using stigmergy to influence the level of exploration and exploitation carried out by an MAS.	33
3.1	Tracking performance of the swarm robotic system.	62
3.2	Response of the swarm robotic system to a dynamic target.	63
4.1	Swarm tracking performance results with and without agent-based memory obtained from the robotic units.	83

List of Abbreviations

CMOMMT	Cooperative Multi-Robot Observation of Multiple Moving Targets
EED	Exploration-Exploitation Dynamics
IMU	Inertial Measurement Unit
MAS	Multi-Agent System
MARL	Multi-Agent Reinforcement Learning
MIPS	Motility-Induced Phase Separation
MRS	Multi-Robot System
PSO	Particle Swarm Optimisation

List of Symbols

a_R	Repulsion Strength
c	PSO social weight coefficient
d	Repulsion exponent
k	Degree of connectivity
k_f	Degree of connectivity of fast agents
k_s	Degree of connectivity of slow agents
L	Length of one side of the square simulation environment
N	Number of agents in swarm
N_f	Number of fast agents in swarm
N_s	Number of slow agents in swarm
$\mathbf{p}$	Point of attraction
r	Random number where $r \in [0, 1]$
r_{ij}	Distance between two agents, i and j
$\mathbf{r}_{ij}$	Vector from agent i to agent j
S_i	Tracking state of an agent i
T_f	Duration of simulation
t_{mem}	Memory length
$\mathbf{u}_i$	Velocity fluctuation of agent i
v_{att}	Attraction velocity
$v_{f,\text{max}}$	Maximum velocity of fast agent
v_i	Velocity of agent i
$v_{o,\text{max}}$	Maximum velocity of target
v_{rep}	Repulsion velocity
$v_{s,\text{max}}$	Maximum velocity of slow agent
$\mathbf{x}_i$	Position of agent i
Γ	Cumulative Velocity Fluctuation Magnitude
Θ	Exploitation Ratio
Λ	Exploration Ratio
Ξ	Tracking Performance
ρ	Swarm Density
ϕ	Heading-Bearing Correlation
ω	PSO Inertia Coefficient

In Memory of
Dr. Kwa Soon Bee & Mrs. Lucy Kwa

Chapter 1

Introduction

Nature has always served as a good source of inspiration for solutions to the problems faced by mankind. The field of swarm intelligence is no exception. Swarm intelligence is the collective behaviour a system derived from the decentralised interaction and responses of multiple agents (Beni and Wang, 1993). While this definition was first given in the context of cellular robotic systems, it can also be used to describe the behaviour observed in natural swarms such as colonies of ants, flocks of birds, and schools of fish. The emergent properties arising from the collective behaviour of a swarm allows the system to achieve goals that an individual swarming agent would find impossible to achieve alone.

A classic example of emergent behaviours in natural swarms are starling murmurations, where flocks of starlings have been observed to display complex kinetic patterns when each bird follows just three simple rules regarding attraction, alignment, and repulsion (Ward and Webster, 2016). In an attempt to replicate this, Reynolds (1987) simulated a flock of birds in a virtual multi-agent model. To do so, each agent within the simulated flock was instructed to avoid collision with its neighbours (repulsion), match its velocity with its neighbours (alignment), and attempt to stay close to its neighbours (attraction). This is similar to the self-propelled particle model proposed by Vicsek et al. (1995), more widely known as the Vicsek Model, who also considered the addition of noise in the form of random velocity perturbations into the model.

Whether artificial or naturally occurring, all swarms tend to be characterised by three main features (Bouffanais, 2016): (1) the lack of a central controller dictating the movements of the individual agents, (2) the ability of agents to perceive local features of the environment that is supplemented through the use of information sharing between agents and, (3) the ability of the system to respond and adapt quickly to a dynamic environment. These features endow swarming systems with unique distributed problem solving capabilities, making swarms an increasingly popular tool to solve many different classes of problems, ranging from computational optimisation in the virtual realm, to target search missions using physical robotic swarms in the real world.

1.1 Swarm Intelligence Algorithms

Currently, the most widespread use of swarm intelligence is in computational optimisation. To that end, myriads of metaphor-based metaheuristics have been developed (Sörensen, 2015), the most well known of which are the Ant Colony Optimisation (ACO) (Dorigo, Birattari, and Stutzle, 2006) and the Particle Swarm Optimisation (PSO) (Kennedy and Eberhart, 1995) algorithms. Such metaheuristics are examples of multi-agent systems (MAS) and often involve the use of a virtual swarm of agents, also known as candidate solutions, that attempt to construct a solution to the given optimisation problem over a set number of iterations. The quality of a solution obtained by an agent is exchanged with its neighbouring agents via some means of communication. Agents then respond to these information exchanges by favouring exploitation of the search space surrounding the solution of the highest fitness.

These metaheuristics have been successfully applied in various problems, ranging from transport network planning (Babazadeh, Poorzahedy, and Nikoosokhan, 2011) to the optimisation of hyperparameters of deep learning models (Qolomany et al., 2017). However, similar to other optimisation metaheuristics, a common problem faced by these types of algorithms is the exploration-exploitation dilemma. This dilemma stems from the decision a system must make when choosing between whether to search the environment for better options (exploration) or to make use of the information that it currently has (exploitation). Exploratory and exploitative actions tend to be on two opposing ends of the same spectrum and the failure to find a balance between the two will result in suboptimal system performances—over-exploration results in the system being unable to find global maxima or minima, while over-exploitation results in the system being trapped by local optima. As such, there is a large body of research dedicated to methods altering the exploration and exploitation characteristics of these virtual swarms.

1.2 Swarm Robotics

A natural extension of swarm intelligence is their application in the domain of robotics gave rise to swarming multi-robot systems (MRS) and the increasingly popular field of swarm robotics. Such systems have been applied in a wide variety of tasks and missions such as area mapping (Kit et al., 2019; Sundram et al., 2019), decentralised decision making (Ebert et al., 2020), area monitoring and defence (Zoss et al., 2018; Shishika and Paley, 2019), and target monitoring and capture (Zhang et al., 2019a; Esterle and Lewis, 2020).

As defined by Şahin (2005), the concept of swarm robotics can be defined by the following distinguishing features:

- Autonomous robots - Physical mobile agents that are capable of physically interacting with the natural environment. Such agents should also be able to make decisions regarding its movements and actions independently.
- Large number of robots - It is generally accepted that a swarm should have a minimum of 10 to 20 agents. Despite this, there is no strictly defined lower bound

for the number of agents within the swarm, meaning that it is possible for a 'swarm' to consist of a small number of robots. Regardless of the swarm size, Şahin stresses the need to aim for scalability of the swarm and that the control algorithm needs to be able to cope when operating within a larger system.

- Homogeneity - A swarm of robots needs to consist largely of the same physical capabilities. While a swarm can consist of multiple homogeneous groups, the number of robots in each group should be large. Systems consisting of agents with a large variety of physical embodiments or behaviours, such as those used in robo-football teams, are not considered to be robotic swarms under this definition.
- Relatively inefficient or incapable robots - Each agent should be incapable or relatively inefficient at performing an assigned task by itself. As such, a robot needs to operate within a swarm where the cooperation between robots of the system is essential to effectively carry out a task.
- Local sensing and communications - Each agent within the swarm can only observe the features of its operating environment in the agent's immediate vicinity. In addition, each is unable to communicate with the entire swarm, ensuring that the coordination of actions between agents remains decentralised.

Given these features, swarming MRS have three main advantages over their centrally controlled counterparts (Dorigo, Theraulaz, and Trianni, 2021):

1. Scalability - The capacity of a system to continue functioning properly when the number of its components substantially vary.
2. Flexibility - The capacity of a system to solve problems/perform tasks that depart from those chosen at design time.
3. Robustness - The capacity of a system to continue working efficiently in environmental conditions different from those considered at design time.

These attributes stem from the lack of a central controller that dictates the actions of the individual agents. 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 react quickly to changes in their surroundings (flexibility), and removes a central point of control, allows the system to continue operations despite agent failures (robustness) and eliminates potential processing bottlenecks (scalability).

In addition to these three widely accepted advantages, Vallegre et al. (2018) also showed the advantages of swarming MRS brought on by being modular systems. The MRS's modularity allows for various upgrades and improvements to be trialled within the system in an iterative fashion. This can be done without having to apply the upgrade to the entire swarm potentially resulting in cost savings.

1.3 Swarm Communications

As just stated, one of the defining features of a robotic swarm is the ability of its agents to communicate with its neighbours in a decentralised fashion. This is a crucial component in facilitating the coordination of agent actions, thereby allowing the MRS to display emergent collective behaviours and accomplish tasks that are greater than the sum of the system's individual components' abilities. In robotic swarms, these communications can take the form of messages being transmitted directly from agent to agent, or can be passed between agents via signals each agent leaves behind in the environment in a process known as stigmergy. Using direct communications, robots are able to send relevant information to their neighbours, such as their pose or measurements of the local environment, mainly through radio communications (Thenius et al., 2016; Kuan, 2018). In contrast, stigmergy is a form of indirect communication inspired by ants where each agent shares information by changing the local characteristics of its environment (Nedjah and Junior, 2019). This is most commonly done by leaving behind markers through the use of tags (Tang et al., 2019) or by manipulating features of the environment (Bayindir, 2016).

When using direct methods of communication, while it might be intuitive to enable each agent to interact with as many other agents as possible, this is not necessarily always the case. In natural starling flocks, it has been shown that each individual bird takes its cues from its 7 nearest neighbours instead of all birds within visual range (Ballerini et al., 2008). Similarly, it has been shown that the number of neighbours within an agent's communication network affects the overall level of responsiveness of a swarming system to its environment, with very low and very high levels of connectivity being associated with low levels of system response (Mateo, Kuan, and Bouffanais, 2017; Mateo et al., 2019).

1.4 Target Search and Tracking

One of the more popular use cases for swarm robotics is in the field of looking for and tracking one or multiple targets. Search and track algorithms can be applied in a wide variety of scenarios, from civilian applications, such as search and rescue (López et al., 2017; Wang and Meghjani, 2020) and environmental monitoring (Coogle and Howard, 2013), to defence related purposes, such as in the detection and clearing of underwater mines (Tan and Bishop, 2004) and protecting a designated area from intruders (Shishika and Paley, 2019; Alvarenga, Basilico, and Carpin, 2020). Due to the diversity of these applications, many studies have been done in the context of several different frameworks, such as the Cooperative Multi-Robot Observation of Multiple Moving Targets (CMOMMT) framework established by Parker and Emmons, 1997, or the similar k -coverage problem established by Huang and Tseng (2005). Most of these works focus on the response of individual agents to the presence of targets in its proximity or the proximity of neighbouring agents (Khan, Rinner, and Cavallaro, 2018; Esterle and Lewis, 2020). In addition to this, most works focus on the cases where targets travel as fast as, or slower than the swarming agents. This is because the studies make the assumption that an fast-moving evasive target will always be able to outmaneuver its pursuers, making the task impossible and therefore trivial (Parker and Emmons, 1997;

Parker, 2002). While Janosov et al. (2017) has shown in pursuit and evasion games involving swarms that the capture of fast-moving targets is indeed possible, this study was performed in the case where the agents had vision over the entire environment. In addition, studies have not been conducted that look at the continued tracking and pursuit of such targets.

1.5 Research Objectives and Approach

Given the lack of work done focusing on the search and tracking of fast-moving targets, the main aim of this thesis is to answer the following research question:

How can an decentralised MRS be made to find and track a target that can move faster than any of its component robots?

When studying an MRS carrying out such a task, there are also many other factors to consider that affect its tracking performance, such as the target's speed and type of movement policy. In addition, there is a host of complications that may arise from the incremental upgrade of swarming agents that use of heterogeneous swarms (Altshuler, 2009; Vallegria et al., 2018; Bennett, Bullock, and Lawry, 2021). These factors that affect a system's performance and the use of heterogeneous swarms will also be explored over the course of this thesis.

In light of these issues, this thesis aims to answer the following sub-questions:

- How can the amount of exploration and exploitation carried out by the swarm be balanced?
- What factors affect a swarm's optimum balance between exploration and exploitation?
- Under what conditions can a heterogeneous swarm be employed in a target search and tracking mission without sacrificing system performance?

It should be noted that the heterogeneous agents used in this work all have the same functionality and capabilities, with the exception of their maximum speeds. As such, the heterogeneous MRS considered in this work still fall under the definition of swarm robotics provided by Şahin (2005). However, it should also be pointed out that there are also a growing number of truly heterogeneous swarms comprised of agents with vastly different abilities, such as the subCULTron swarm by Thenius et al. (2016) and Swarmanoid by Dorigo et al. (2013).

1.6 Contributions

Given the research questions to be answered, the main contributions of this thesis are summarised as follows:

- A novel search and track strategy that enables both homogeneous and heterogeneous MRS to find and pursue fast-moving targets—targets that move faster than their pursuing agents—in a scenario with limited agent sensing range. The targets to be tracked can move in both evasive or non-evasive patterns. This is done

in a decentralised fashion and is achieved by using local force vectors to control the movement of agents. The overall system's exploration-exploitation dynamics (EED) is controlled through the adjustment of three parameters: (1) the value of k of an interconnecting k -nearest neighbour communications network, (2) the length of an agent's memory, t_{mem} , and (3) an adaptive repulsion parameter that controls an agent's level of repulsion from its neighbours. This strategy is first developed and tested within simulated environments before being validated by its implementation within a physical MRS.

- A set of metrics that allow the quantification of a system's exploration-exploitation balance. While it is generally agreed that level of exploration and exploitation carried out by a virtual swarm in optimisation problems can be characterised by the amount of spread of the candidate solutions (Zhan et al., 2009), there is currently no widely agreed upon metric that measures this quantity in physical MRS. As such, a set of new metrics are also introduced in this thesis to allow the study of a system's swarm intelligence and quantify its EED.
- A study on the factors that affect a system's EED balance. The work done in this thesis shows that during the operation of an MRS, there is an optimum EED balance at which a system's performance is maximised. This balance is influenced by several parameters that can change the system's EED balance. Changes to these factors can serve to bias the MRS towards carrying out a higher level of exploratory or exploitative actions, driving the system away from its optimum EED balance. Here in this thesis, the influence of some of these factors, namely: the movement profile of the target(s), the number of targets present, the speed of the targets, the agent density of the MRS, and the size of the MRS, are studied. Analysing these effects will allow a swarm designer to adequately choose the correct system parameters for their swarm, thereby maximising its performance.

1.7 Thesis Outline

The rest of this thesis is structured as follows:

- Chapter 2 introduces the concept of exploration and exploitation in a dynamic environment while using multi-agent systems. The aim of the chapter is to provide a summary of the methods that can be used to control the balance between the two activities. Given the range of activities in which MAS can be used, this chapter focuses on controlling the exploration-exploitation balance in the context of target search and tracking, as well as various related problems in computational optimisation.
- Chapter 3 introduces the concept of target search and tracking. A basic swarm strategy enabling a swarming MRS to carry out target search and tracking is introduced, along with several metrics that allow for the quantification of a swarm's EED while carrying out a target search and tracking task. The strategy, which employs a variable k -nearest neighbour network to vary the systems exploration-exploitation balance, is implemented within a system tasked with searching for

and tracking non-evasive targets while utilising swarms consisting of homogeneous and heterogeneous agents. The results obtained in simulations for the homogeneous swarm are also validated using a physical MRS tracking a virtual target.

- Chapter 4 further builds on the strategy presented in Chapter 3 to adapt it for use in tracking evasive targets. This is done through the introduction of agent-based memory, which is another parameter that can be used to tune a swarm's EED balance. This updated strategy is tested using both homogeneous and heterogeneous systems tasked with tracking fast-moving evasive targets. The results and observations made in simulations for the homogeneous case are corroborated with those gathered from physical MRS experiments.
- Chapter 5 studies how the performance of both homogeneous and heterogeneous systems varies according to the agent density. This study reveals three distinct phases with varying tracking performances that correspond to a different swarm densities and the influence of this density on a system's ability to carry out exploration and exploitation. The factors that affect when the three phases occurs are also discussed in this chapter.
- Chapter 6 closes this thesis with a summary and a discussion on the results presented in this work. The prospects of future work stemming from the findings laid out in this thesis are also discussed.

Chapter 2

Balancing Exploration and Exploitation

2.1 Introduction

The balance between exploration and exploitation is a crucial element in a target tracking task. This dilemma arises owing to the fact that gathering new information (i.e., exploration) and making use of currently available information (i.e., exploitation) tend to be two mutually exclusive activities (Berger-Tal et al., 2014). Should a system favour exploratory actions, it would be able to gather large amounts of knowledge without being fully able to benefit from this knowledge. This may increase the time required for the MAS to accomplish its task or prevent it from achieving its assigned task altogether (Dadgar, Jafari, and Hamzeh, 2016). On the other hand, should a system favour exploitative behaviours, the system may be unable to adapt to changes should it be deployed in a dynamic environment. Such exploitation-focused systems may find also find themselves trapped in a local optimum if deployed in static environments (Azoulay-Schwartz, Kraus, and Wilkenfeld, 2004; Berger-Tal et al., 2014; Mehlhorn et al., 2015). Despite this apparent trade-off between the two activities, both sets of actions can be performed simultaneously (Stadler, Rajwani, and Karaba, 2014). As such, one can expect that to maximise the performance of such a system, an optimum balance, which is possibly a dynamic balance, between the amount of exploration and exploitation carried out needs to be attained.

Given the importance of exploration–exploitation dilemma in MAS, this chapter focuses on introducing the dilemma in the context of MAS performing a target search or target tracking task, as well as various related scenarios, such as computational optimisation and resource foraging (tasks that both involve searching for points of interest). In addition to this, it is shown how various agent actions can be classified as either exploratory or exploitative ones. The methods used by MAS practitioners to measure the balance between exploration and exploitation as well as the methods that are currently

The content in this chapter has been published in Kwa, H.L., Kit, J.L., and Bouffanais, R. (2022). “Balancing Collective Exploration and Exploitation in Multi-Agent and Multi-Robot Systems: A Review”. In: *Frontiers in Robotics and AI*. 8:771520 DOI: 10.3389/frobt.2021.771520

used to control this balance are also summarised. Readers already familiar with these concepts should feel free to skip this chapter.

2.2 Characterising Exploration and Exploitation

The concept of exploration and exploitation is a broad and multi-disciplinary one and is used in a variety of fields and contexts. As such, even within the field of target search and tracking, there is no unique definition that is used to commonly define these two behaviours. Nevertheless, these exploratory and exploitative actions can be characterised by certain observable features. To this end, Mehlhorn et al. (2015) identifies three key distinct features that distinguish between the two: (1) behavioural patterns, (2) the uncertainties associated with an actor's choices, and (3) the expected outcomes as a result of an agent's actions. It is important to note at this point that while these three features classify the actions of a single agent, a single agent's actions do not characterise the exploration–exploitation dynamics (EED) of the entire MAS. It is only through observing the actions of all or a proportion of a system's agents that one is able to accurately characterise the exploration–exploitation balance of an MAS.

An agent's behavioural patterns are one of the most common definitions of exploration and exploitation used as the movements it carries out is a physical quantity that can be directly observed and measured. Many MAS practitioners use the agents' movements and observed positions to determine the system's exploration-exploitation balance. For example, in a target search context, when agents are close together or traveling towards the same region of the area of operations, it can be assumed that they are exploiting a common source of information (Jordehi, 2014). Conversely, if agents are moving away from each other or in a random fashion, they can be assumed to be carrying out exploration.

The less tangible aspect of an agent's behavioural patterns are its intentions that drive the observed actions. While not as obvious as an agent's motions, these can still be easily determined by querying the state of an individual agent in an engineered MAS. Stadler, Rajwani, and Karaba (2014) posited in their review that exploration and exploitation are associated with a certain level of learning. Therefore it can be said that an agent is carrying out exploitation if it uses previously acquired knowledge and carrying out exploration if it is not doing so (Gupta, Smith, and Shalley, 2006; Nauta, Simoens, and Khaluf, 2020). Again, this is another common method of defining exploration and exploitation in MAS, where certain practitioners have developed dedicated behaviours that facilitate exploration and exploitation separately.

Besides observing an agent's behavioural patterns, the amount of uncertainty associated with each action can also be used to classify exploration and exploitation. Through this manner of characterisation, exploratory actions are associated with those that have high levels of uncertainty, while exploitative ones are associated with low levels of uncertainty. In MAS operating in unknown environments, these uncertainties are usually presented in the form of a map that updates itself as the task progresses (Wolek et al., 2020; Crosscombe and Lawry, 2021; Liu, Crosscombe, and Lawry, 2021). Using such maps allows an agent to move towards regions of different levels of uncertainty based on its intentions to carry out either exploration or exploitation. As such, an agent that

moves towards or gathers information in an area with high uncertainty is deemed to be carrying out exploration, while an agent doing the same in an region of low uncertainty can be said to be carrying out exploitation.

Finally, the type of action an agent performs can be classified based on the expected results of the said action. This is typically exemplified by observe-or-bet tasks. In such a task, an agent is given a chance to ‘observe’ the environment and gain information while not receiving any reward (exploration). Alternatively, the agent may ‘bet’ on an option and receive its associated reward (exploitation), although the amount of reward received remains hidden from the agent (Tversky and Edwards, 1966; Navarro, Newell, and Schulze, 2016). While these tasks offer a very clear distinction between exploration and exploitation, such contrasts are not as explicit in the study of real systems; agents tend to know the quantity of a reward and the action that was performed that led to the reward. These additional pieces of knowledge allow an agent to obtain more information about its surrounding environment and allow agents to simultaneously perform exploration and exploitation (Gupta, Smith, and Shalley, 2006).

As highlighted by Mehlhorn et al. (2015), these three characteristics need not be used to define exploration and exploitation in isolation; two or more can be used to determine the type of activity being carried out by an agent. For example, when searching for environmental features in an unexplored environment, Matignon and Simonin (2018) proposed using an MRS that built an occupancy grid map that increased in resolution as the task progressed. If a robot chose to move into a previously un-visited cell, which is an action associated that is with learning and a high level of uncertainty, it was determined to be exploring. If the robot chose to move to a cell with a known object of interest or update its map of the local area with a higher resolution, an action associated with obtaining a reward and a low uncertainty, it was determined to be exploiting.

2.3 Measuring the Exploration-Exploitation Balance

Before the exploration-exploitation balance of a system can be adjusted, appropriate metrics need to be defined to measure and quantify the balance between the two opposing activities. Like the definition of exploration and exploitation, these metrics are task-dependent and there is no unique quantity that has been used to characterise the exploration-exploitation balance of different systems.

In the field of computational optimisation, a typical strategy in finding a global optimum is through the use of an MAS consisting of many candidate solutions searching the solution space. In this use case, the quantification and monitoring of the system’s EED is important as doing so permits it to autonomously detect if the solutions have been trapped in a local optimum or have stagnated. Doing so grants the system the ability to autonomously modify several specified parameters, allowing it to “jump out” and continue with its optimisation task (Palmieri and Marano, 2016; Bonyadi and Michalewicz, 2017). Similarly in MAS operating in dynamic environments, these measurements also allow a system to monitor and prevent solution diversity loss, allowing systems to track moving optima over time (Blackwell and Branke, 2006; Leonard, Engelbrecht, and Wyk, 2011; Jordehi, 2014). At this point, it should be noted that a

system's diversity is analogous to the amount of exploration it carries out. The metrics used by the studies covered in this review largely fall within the categories of spatial distribution based metrics or probability based metrics, although there are also other metrics that have been used that do not fall within these two categories.

While most exploration and exploitation metrics are measured globally, i.e., a system-wide property that allows one to quantify a system's EED as a whole, some of these metrics can also be measured locally by the individual agents, thereby allowing them to adjust and adapt their actions appropriately. The use of globally and locally computed metrics both have their associated advantages and disadvantages. While the use of a system-wide EED metric would allow for a more accurate quantification of the overall balance between exploration and exploitation, it also entails that this balance be controlled in a centralised fashion. This would result in the loss of a certain degree of system scalability, flexibility, and robustness. Conversely, if a locally computed metric is used, the agents would be able to use this metric during their operations, allowing them to influence the behaviour of the system through various feedback or feed-forward loops. While such agent-based metrics may not be able to quantify a system's EED as accurately as a global metric, it allows the system to maintain a level of decentralisation and hence, scalability, flexibility, and robustness.

2.3.1 Spatial Distribution Based Metrics

Similar to classifying an agent's actions based on its behaviour and movement patterns, the use of metrics based on the agents' spatial distribution is very popular as it is a physical quantity that can be easily observed, measured, and calculated. As previously mentioned, to quantify the level of exploration and exploitation in computational MAS, the solution diversity of a system is usually calculated. This solution diversity is a measure of the spatial distribution of all candidate solutions within the feasible region. Cheng et al. (2015) based their diversity metric on the average distance of each candidate from the mean solution. Hussain et al. (2019) adapted this metric for use in their study of swarm-based metaheuristic algorithms by using the median solution instead of the mean as it reflected the population centre of the candidate solutions more accurately. In a similar fashion, Zhan et al. (2009) calculated a *population standard deviation* to measure the solution spread. In essence, all these metrics are essentially based on low-order indicators of the statistical distribution of all the candidate solutions.

Comparable metrics are in use to measure the amount of exploration and exploitation carried out by agents within MRS. As seen with the metrics used in computational optimisation, measuring the the spatial distribution of agents is also used to quantify the level of exploration and exploitation. In their target search task, Dadgar, Couceiro, and Hamzeh (2017) and Dadgar, Couceiro, and Hamzeh (2020) measured the diversity of individual robots within their MRS by calculating the amount of overlap between a robot and its neighbours' search areas. Agent sensor area overlap was also used by Sardinha, Dragone, and Vargas (2020a) who calculated the size of the area covered by at least two agents as a measure of the amount of effective exploration performed by their swarm in a surveillance task. This metric is similar to the one used by Esterle and Lewis (2017), Esterle and Lewis (2020), and Frasheri, Esterle, and Papadopoulos (2020)

who used a k -coverage metric that calculated the amount of time a target spent being tracked by at least k agents.

Stogiannos, Alexandridis, and Sarimveis (2020) measured the level of exploration and exploitation by tracking three metrics: (1) the minimum distance between agent pairs, (2) the sum of the distances between detected targets and their closest agents, and (3) the distance moved by each agent over the course of one time-step. While often not explicitly stated, the distance between a robot and its neighbours is the most common measure of diversity as most MRS control methods tend to focus on preventing excessive aggregation or crowding in small areas (Hereford and Siebold, 2010; Meyer-Nieberg et al., 2013; Lv et al., 2016; Meyer-Nieberg, 2017; Chen and Huang, 2019; Coquet et al., 2019; Coquet, Arnold, and Bouvet, 2021).

The lack of distance measurements does not preclude the use of spatial distribution metrics. As a proxy for the amount of exploitation performed by each robot, Yang, Wang, and Bauer (2019) calculated a communications-based aggregation degree from the number of neighbours within visual range, with higher numbers of agents within visual range suggesting a higher amount of aggregation. This eliminates the need to share positional information between agents, allowing an agent to reduce the amount of information an agent needs to send during each radio transmission. In an adaptation to the BEECLUST algorithm, which was first proposed by Kernbach et al. (2009) and Schmickl et al. (2009), Wahby et al. (2019) determined the level of local aggregation, and therefore the level of exploitation of individual agents, by measuring the time interval between two consecutive neighbouring robot encounters.

The aggregation degree may not necessarily need to be calculated with the use of spatial information. In performing a search for multiple targets using several sub-swarms comprised of leaders and followers, Tang et al. (2020b) calculated the aggregation of the sub-swarms by comparing the fitness value of each sub-swarm leader to the mean fitness value of all the leaders within the swarm. This is similar to the aggregation degree proposed by Yang et al. (2007), who calculated the ratio between the agent with the lowest fitness value and the agent with the highest fitness value. A list of works using spatial metrics to quantify an MAS' EED can be found in Table 2.1.

2.3.2 Probability Based Metrics

While less popular than the use of spatial distance metrics, the probability of actions carried out by individual agents can also be used to quantify the level of exploration and exploitation involved. The most common use of probability based metrics can be found in the application of Lévy walks and flights. Lévy walks are a class of random walks where the distance of the walk, l , is taken from a heavy-tailed probability distribution function, $P(l)$. This probability distribution function is given by the equation:

$$P(l) \sim l^{-\mu}, \quad (2.1)$$

where μ is the Lévy parameter controlling the shape of the probability distribution function. Using this probability distribution function, lower values of μ close to unity

Reference	Metric	Task
Sardinha, Dragone, and Vargas, 2020a	Area observed by at least two agents	Area Coverage
Blackwell and Branke, 2006; Blackwell, 2007; Hereford and Siebold, 2010; Leonard, Engelbrecht, and Wyk, 2011; Jordehi, 2014	Euclidean distance between solutions	Optimisation
Cheng et al., 2015	Euclidean distance between solution and mean system solution	Optimisation
Hussain et al., 2019	Euclidean distance between solution and median system solution	Optimisation
Zhan et al., 2009	Standard deviation of all candidate solutions	Optimisation
Tang et al., 2020b	Fitness value between sub-swarm leaders and mean fitness value	Optimisation
Schmickl et al., 2009; Wahby et al., 2019	Local agent density	Optimal Area Aggregation
Palmieri and Marano, 2016; Dadgar, Jafari, and Hamzeh, 2016; Dadgar, Couceiro, and Hamzeh, 2017; Dadgar, Couceiro, and Hamzeh, 2020	Agent sensor overlap region	Target Search
Meyer-Nieberg et al., 2013; Lv et al., 2016; Meyer-Nieberg, 2017; Chen and Huang, 2019; Coquet et al., 2019; Coquet, Arnold, and Bouvet, 2021	Euclidean distance between agents	Target Search
Yang, Wang, and Bauer, 2019	Number of agents within visual range	Target Search
Esterle and Lewis, 2017; Esterle and Lewis, 2020; Frasheri, Esterle, and Papadopoulos, 2020	k -Coverage	Target Tracking
Stogiannos, Alexandridis, and Sarimveis, 2020	Distance between agents, distance between targets and agents & distance moved by agents in one time-step	Target Tracking

TABLE 2.1: List of works using spatial distribution metrics to quantify MAS exploration and exploitation.

lead to a higher probability of longer walk lengths between direction changes, which is associated with higher levels of exploration. Conversely, higher values of μ cause an agent to preferentially carry out shorter walks between changes of direction, characteristic of higher levels of exploitation (Viswanathan et al., 2000; Viswanathan, Raposo, and Luz, 2008). The Lévy parameter has been used as a method of quantifying and controlling the level of exploration and exploitation of various MRS applied in different tasks, ranging from foraging (Zedadra et al., 2019; Nauta, Khaluf, and Simoens, 2020; Nauta, Simoens, and Khaluf, 2020; Nauta et al., 2020), to target search (Senanayake et al., 2016; Harikumar, Senthilnath, and Sundaram, 2019; Pang et al., 2019), to area mapping (Ramachandran, 2018; Kegeleirs, Ramos, and Birattari, 2019; Ramachandran, Kakish, and Berman, 2020). Methods that employ changes of the Lévy parameter will be explored later in Section 2.4.1.

In addition to the use of Lévy distributions, action probabilities can be used to set the level of exploration and exploitation carried out by an MRS. To search for a target, Matignon and Simonin (2018) used action probabilities to bias an agent in favor of carrying out either exploratory or exploitative actions. In a similar fashion, Falcón-Cortés, Boyer, and Ramos-Fernández (2019), Nauta, Simoens, and Khaluf (2020), and Nauta, Khaluf, and Simoens (2020) uses the probability of an agent using information in its memory as means of determining and controlling the level of exploration and exploitation of the system. A list of works using such probability based metrics can be found in Table 2.2 for different tasks.

Reference	Metric	Task
Ramachandran, 2018; Kegeleirs, Ramos, and Birattari, 2019; Ramachandran, Kakish, and Berman, 2020	Lévy parameter	Area Mapping
Zedadra et al., 2019; Nauta et al., 2020	Lévy parameter	Resource Foraging
Falcón-Cortés, Boyer, and Ramos-Fernández, 2019; Nauta, Khaluf, and Simoens, 2020; Nauta, Simoens, and Khaluf, 2020	Lévy parameter and memory utilisation probability	Resource Foraging
Viswanathan et al., 2000; Viswanathan, Raposo, and Luz, 2008; Harikumar, Senthilnath, and Sundaram, 2019; Pang et al., 2019	Lévy parameter	Target Search

TABLE 2.2: List of works using probability based metrics to quantify MAS exploration exploitation dynamics.

2.3.3 Other Metrics

In addition to the metrics based on spatial distribution and probability just discussed, there are a few other metrics used by various works that do not fit into these two broad

categories. For example, when carrying out a search for multiple targets under the Cooperative Multi-Robot Observation of Multiple Moving Targets (CMOMMT) framework, Banfi et al. (2015) introduced a metric known as the *tracking fairness* that measures the monitoring effort of the MRS across all targets. Should a target receive more attention from the MRS' agents, the system would have a high tracking fairness score, indicating over-monitoring of a target or too much exploitation being carried out by the system. Conversely, a low fairness score indicates that the agents are monitoring all the targets evenly and that there is a good balance between exploration and exploitation.

To develop a tracking strategy for the same CMOMMT problem using deep reinforcement learning Yan, Jia, and Bai (2021) used an occupancy grid map (OGM) to characterise the environment. A tracking fairness based metric was then used, comparable to the one proposed by Banfi et al. (2015), as well as an exploration metric based on the latest observation time of all grid cells in the OGM.

The metrics used to measure the exploration-exploitation balance presented in the later chapters of this thesis is also unable to be classed within the umbrellas of spatial distribution and probability based metrics. In this work, two metrics are proposed. The first is based on an agent's direction of travel and its bearing to the target. This allows for an individual agent's contribution, and hence the system's overall exploration and exploitation dynamic to be quantified. Should an agent's position relative to a target be unable to be determined (e.g., when tracking multiple targets), a second metric is proposed based on a swarm's *engagement ratio*. The use of this ratio allows for the quantification of a swarm's EED in the presence of multiple targets by finding the proportion of agents actively moving towards a target. These two metrics will be further discussed in Chapter 3.

2.4 Agent Response Methods

With the quantification of a system's exploration-exploitation balance, this balance between the two activities can now be controlled. The most common method of adjusting this balance is by changing an MAS agent's actions in response to locally measured variations in the environment or information originating from the agent's neighbours. This method of controlling exploration and exploitation can elicit two types of response from an agent: (1) trigger a discrete change in state of an agent, causing it to perform a predefined set of actions associated to either exploration or exploitation, or (2) cause a small change in an agent's actions, resulting in a small change in bias of the agent's actions towards either end of the exploration or exploitation spectrum. A visual representation of the types of agent response methods used can be found in Fig. 2.1. Of the methods shown in the figure, methods involving area-task assignments will not be covered in this thesis and was only included for completeness. This is because such methods are primarily used in scenarios where an MAS carries out distributed task allocation, which is not the focus of this work. Similarly, strategies based on the Artificial Immune System (AIS) will not be covered as they are also primarily used for task assignment scenarios.

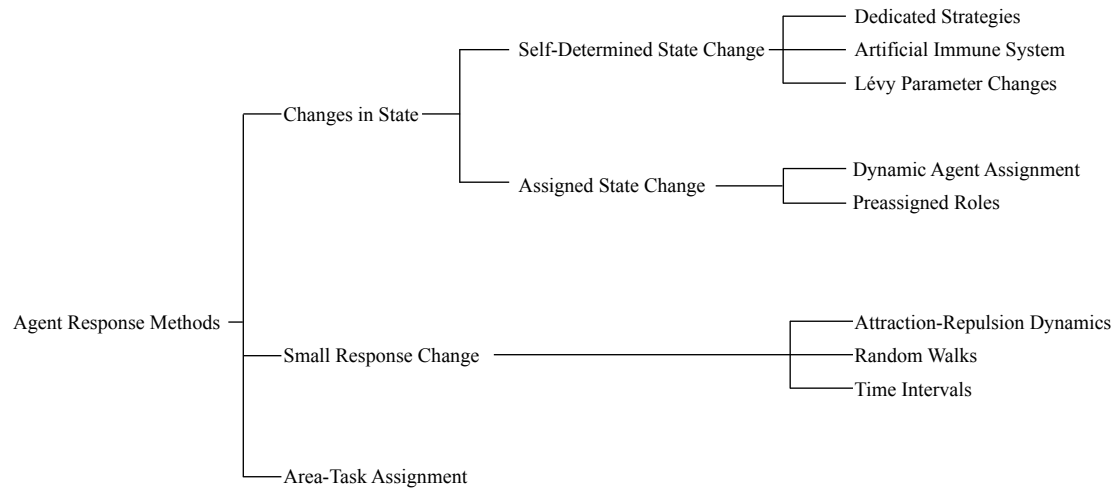


FIGURE 2.1: Classification of different agent response methods to control an MAS' exploration-exploitation balance.

2.4.1 Changes in State

When deploying changes in agent states, a large change in an agent's behaviour is effected in response to a given stimulus observed by either an agent, a subset of agents, or the entire MAS itself. This gives agents the ability to switch from an exploratory to an exploitative behaviour, and vice-versa allows for them to better adapt to their local environment, with the aim to improve the overall performance of the system. These strategies tend to base their exploration strategies on random walks (Zedadra et al., 2017), followed by information exploitation driven by either a metaheuristic algorithm or direct travel towards a point of interest should one be detected. A flowchart of such changes in agent response can be seen in Figure 2.2.

These changes in response can either be triggered by an agent itself as a reaction to its local environment or a coordinating agent can trigger this response change in a certain subset of the MAS. The latter set of strategies usually involves a centralised controller or locally designated leader to determine which agents are to change their response patterns. As can be expected, each set of strategies comes with its own advantages and disadvantages. For instance, while using decentralised strategies, an MAS is able to retain its flexibility, robustness, and scalability. However, keeping these properties of the system comes at the expense of more precise control over the overall system's EED. Using a set of strategies involving a centralised controller, an MAS user is able to exert greater and more accurate control over the system's EED. However, this trade-off must be found, which inevitably impacts one of the system's three key characteristics of flexibility, scalability, and robustness.

Self-Determined State Changes

In decentralised systems, changes in an agent's state stem from its own reaction to changes in its local environment. Based on this information, an agent decides for itself if it will carry out exploratory or exploitative actions. When determining their own

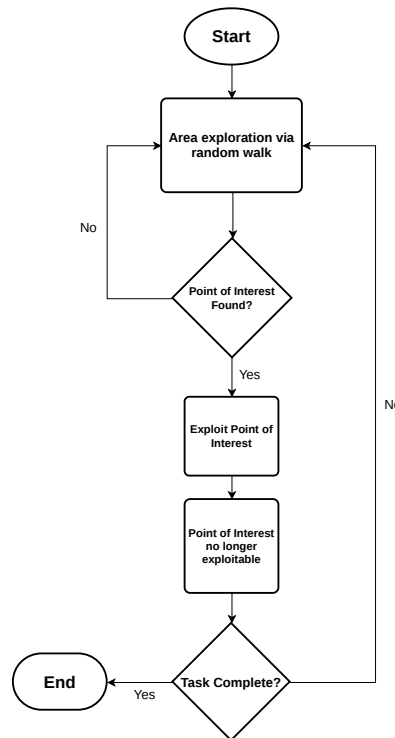


FIGURE 2.2: Flowchart detailing how an agent deterministically decides on its response when provoked by environmental stimuli.

state in response to their environment, an agent normally takes into account local observations carried out by itself as well as information originating from its neighbours. This allows an agent to react quickly to any changes in its local environment, allowing the MAS to retain its system flexibility. Agents are also not reliant on a leader or central controller assigning it an appropriate response, allowing the system to maintain its scalability and robustness. However, given this decentralised control strategy, the MAS will have less control over the overall amount of exploration and exploitation carried out. A list of such strategies and associated tasks can be found in Table 2.3.

Dedicated Exploration and Exploitation Strategies In various systems, agents may switch between pre-programmed exploratory and exploitative behaviours. To capture multiple static targets, the approach proposed by Li et al. (2014) initialised their agents with a random search behaviour to explore the environment. In most works involving simulations, a target signal is modeled by using an objective value function, representing the type of information to be exploited by the MAS (e.g., radio signal intensity, chemical plume concentration, etc.). When a target signal was encountered, agents would then employ a Particle Swarm Optimisation (PSO) algorithm, allowing the system to exploit the target signal and capture the target. After capturing the target, the agents would revert to their original state and resume their random environment search.

Similar changes in state and strategy have been utilised in searching for and tracking dynamic targets. In the tracking of multiple dynamic objects of interest under the

Reference	Strategy	Task
Hecker, Carmichael, and Moses, 2015	Random walk search phase & agent assignment exploitation phase	Resource Foraging
Nauta, Khaluf, and Simoens, 2020; Nauta, Simoens, and Khaluf, 2020	Lévy Walk exploration & memory driven exploitation	Resource Foraging
Zedadra et al., 2018; Zedadra et al., 2019	Dedicated exploration & exploitation Lévy parameters	Resource Foraging
Li et al., 2014; Zhang, Zeng, and Xue, 2014	Random search & PSO tracking strategy	Target Capture
Kolling and Carpin, 2006; Kolling and Carpin, 2007; Esterle and Lewis, 2017; Esterle and Lewis, 2020	Random search & target following strategy	Target Tracking (CMOMMT)
Fraseri, Esterle, and Papadopoulos, 2020	Pattern search & target following strategy	Target Tracking (CMOMMT)
Harikumar, Senthilnath, and Sundaram, 2019	Dedicated exploration & localisation Lévy parameters	Target Suppression

TABLE 2.3: List of works using self-determined state change strategies to influence a system's exploration and exploitation dynamics.

CMOMMT framework, Esterle and Lewis (2017) and Esterle and Lewis (2020) programmed their agents to initially explore the area randomly. However when an object of interest was detected, it would transition into a tracking mode, change its response, and ultimately alter its velocity to follow that point of interest. The authors were able to demonstrate that this mixed strategy of random motion and following a target yielded better tracking performances than an exploration-only strategy in which the agents only move around randomly.

Changes in state may also be triggered by signals originating from neighbouring agents. In their work tracking multiple targets, Kolling and Carpin (2006) and Kolling and Carpin (2007) explored the use of so-called 'help calls' broadcast from an agent when a target is predicted to be leaving its observation range. Other robots that receive this help call that are not involved in tracking a target move towards the robot requesting help to assist with the tracking task. The use of help calls is also explored by Esterle and Lewis (2017) and Esterle and Lewis (2020) who studied their applications in conjunction with various response and communication models. Fraseri, Esterle,

and Papadopoulos (2020) expanded on this work, incorporating a *willingness to interact* parameter that determined the propensity of an agent to aid in the tracking of a target.

Agents are also able to form dynamic groups dedicated to exploiting information stemming from a point of interest. The formation of groups is more commonly done together by means of a central controller or through a locally designated leader. However, these dynamic groups can still be formed using decentralised decision-making. Zhang, Zeng, and Xue (2014) proposed a strategy to search for multiple targets in which agents are initialised in the random-walk search mode. When a target is found, a dynamic group comprised of a maximum of 10 agents is formed. Agents in this group exploit the target's emitted signal through the use of a PSO algorithm. At each time-step, agents broadcast their measured target signal strengths and individually decide if they should join or leave the group.

Lévy Parameter Changes Instead of using vastly different movement policies to effect changes in the response of agents, agents using Lévy walks can vary their Lévy parameter. As explained in Section 2.3.2, changing the Lévy parameter μ can be used to influence the overall EED of an MAS; higher Lévy parameter values are used to shorten the walk lengths of agents, prioritising exploitation, while conversely lower μ values increase walk lengths and hence favour exploration. As such, several MAS make use of changing their Lévy parameter to influence its EED instead of using dedicated exploration and exploitation algorithms.

In a resource foraging task, Nauta, Khaluf, and Simoens (2020) and Nauta, Simoens, and Khaluf (2020) paired their use of changing Lévy parameters together with agent based memory to exploit resource dense patches in an environment. For a system tasked with locating resources in an environment, Zedadra et al. (2018) and Zedadra et al. (2019) used a time-based system paired with a 'satisfaction' parameter to determine the transition from the exploitation mode to the exploration mode. When exploring, the MAS used a small Lévy parameter to generate long-step sizes. This was changed to favour smaller steps when a target was found to encourage exploitation. While in the exploitation mode, an agent's satisfaction parameter value was increased for each time-step a target was encountered and decreased when no target was met. An agent would only transition from the exploitation to exploration when both the satisfaction value reached zero and the minimum time limit for the exploitation mode was reached.

To determine the location of forest fires, Harikumar, Senthilnath, and Sundaram (2019) proposed using a temperature threshold to determine an agent's state. In their MAS, all agents are initialised in a searching state, using a low μ value to direct area exploration. Should an agent detect a temperature higher than a specified threshold value, it would transition to a more exploitation-focused strategy by employing either a directional-driven Brownian search or regular Brownian search strategy, drawing from a Lévy probability distribution function with a higher μ value.

Assigned State Changes

While self-determined changes in state are mostly found in fully decentralised systems, assigned state changes can be found in fully centralised systems, i.e., systems where the movement of all agents is dictated by a central controlled, and hierarchical systems, i.e., systems with designated 'leader' agents determining the actions of a small proportion of agents. Using assigned state changes, when a stimulus is found, only a specific subset of the system comprised of a fixed number or predefined subset of agents is assigned to exploit the new stimulus information. This is in contrast to the strategies seen in the previous section in which agents are able to decide for themselves if they should switch behaviours. Using such behaviours allows for greater and more precise control over the exploration-exploitation balance of an MAS. However due to the more centralised nature of such strategies, a system must sacrifice a certain degree of flexibility, scalability, and robustness. Using this class of strategies, the assignment of roles and responses to agents can either be done dynamically as the task progresses, or during the initialisation process. A list of works assigning agent response patterns can be found in Table 2.4.

Dynamic Agent Assignment While using assigned state changes does indeed reduce a system's flexibility, a certain degree of flexibility can still be retained by having agents assigned to designated targets while a mission is being carried out. Ni and Yang (2011) demonstrated a dynamic agent assignment strategy when carrying out a target search mission in a 2D environment. In their task, agents initially started performing a random walk to explore the environment. Upon discovering a target, a leader agent is designated to coordinate a dynamic group comprised of four agents. While these agents pursue the discovered target, the rest of the agents continue with exploring the environment. Ni et al. (2018) later built on this work and demonstrated this type of strategy to search for and pursue targets in an underwater 3D environment. The agents employed initially used a Lévy flight-based cruise phase to search for one of multiple targets. When a target is found, a sub-swarm is formed using a centralised self-organizing map (SOM). Only members of this sub-swarm carry out exploitation of the target's signal through the use of the Dolphin Swarm Algorithm (DSA) while the rest of the agents continue exploring the search area. In both strategies, after successfully capturing the target, the pursuing sub-swarm's agents disband and continue with the search process by reverting to the cruise phase. Key to this strategy is the dynamic nature of the sub-swarm's membership, meaning that agents in the sub-swarm may be replaced should it stray too far from the target.

In their work done on tracking and monitoring of multiple dynamic targets with UAVs in a 3D environment, Farid et al. (2018) employed a strategy where all agents carried out a patrol phase based on the Lévy flight. When a target is detected, the agent closest to the target is designated as leader. The leader selects the next two agents that are nearest to the target's location to form a sub-swarm comprising of three agents. Only members of this sub-swarm transition to a PSO-based tracking mode to exploit this information. This pursuit is maintained until the target leaves the surveillance area. Should the original agents move too far away from the target, the sub-swarm leader is able to substitute the original agents with others that are closer to the target.

Reference	Strategy	Task
Sánchez, Vargas, and Couceiro, 2018	Dynamic sub-swarm membership	Area Exploration
Thenius et al., 2016; Hamann, 2018	Dedicated environment exploration and long term monitoring (exploitation) agents	Environment Monitoring
Carlisle and Dozier, 2000; Hu and Eberhart, 2002; Zhan et al., 2009	Forced re-initialisation of a subset of candidate solutions	Optimisation
Blackwell and Bentley, 2002; Blackwell and Branke, 2006; Blackwell, 2007	Dedicated exploratory and exploitative particles	Optimisation
Couceiro, Rocha, and Ferreira, 2011c; Couceiro, Rocha, and Ferreira, 2011a; Couceiro et al., 2014b; Couceiro and Portugal, 2018	Dynamic sub-swarm membership	Target Search
Wang and Meghjani, 2020	Search strategies assigned based on agent's distance to target and target-to-searcher ratio	Target Search
Yan et al., 2018	Forced re-initialisation of robots	Target Search
Wang and Cavallaro, 2016; Schranz and Andre, 2018	Predetermined static agents for target search and mobile agents for target tracking	Target Tracking (CMOMMT)
Cao and Sun, 2017; Cao and Sun, 2018; Ni and Yang, 2011; Ni et al., 2018; Ge, Chen, and Xu, 2018; Farid et al., 2018	Specified number of agents closest to target chosen to utilise a target tracking strategy	Target Tracking

TABLE 2.4: List of works using assigned agent responses to influence a system's exploration and exploitation dynamics.

Similar techniques that assign agents to targets have also been demonstrated by Ge, Chen, and Xu (2018) in aquatic surface operations and by Cao and Sun (2017) and Cao and Sun (2018) in underwater operations. Instead of using a random walk, the authors generate a potential field fitness function based on unexplored areas of the environment and the location of obstacles within the operating area. Exploration is carried out through the use of a PSO algorithm that makes use of this potential field by driving agents towards unexplored regions of the environment. When a target is found, the four agents closest to the target enter a tracking mode and are assigned to pursue that target until it is caught. This is achieved through the use of a PSO algorithm on the target's emitted signal, driving the agents towards the target's location. Upon a successful target capture, these agents revert to the search mode to continue the search for more targets. In another study done on search and rescue strategies by Wang and Meghjani (2020), searcher strategies are assigned to an agent based on the agent's distance to a potential target and the ratio of searchers to targets.

Unlike the works just discussed, size of the sub-swarms need not be fixed. In Couceiro, Rocha, and Ferreira (2011c), Couceiro, Rocha, and Ferreira (2011b), Couceiro et al. (2014b), and Couceiro and Portugal (2018), work was done on a Robotic Darwinian PSO (RDPSO) and adapted for use with ground robots from the original Darwinian PSO by Tillett et al. (2005). Sánchez, Vargas, and Couceiro (2018) later expanded on this work with applications using underwater robots carrying out exploration in a 3D environment. This strategy balances the swarm's EED through social exclusion or inclusion of agents and is done by splitting the MRS into multiple sub-swarms and independent 'wanderers' that search for the solution individually. All of these sub-swarms execute the same target tracking strategy independently and there is no direct information transfer between different sub-swarms. However, there are some system-level rules applied during each iteration that serve to ensure that the sub-swarms' membership remains dynamic, thereby facilitating indirect information transfer between different groups.

In numerical optimisation, where it is easier to control the movement of individual agents, it may be decided that sub-swarms may not be needed and that they should be dispersed. As such, a common strategy observed in this field is to trigger the re-initialisation of particles within the search space (Carlisle and Dozier, 2000; Hu and Eberhart, 2002; Zhan et al., 2009). Using these strategies, a central controller will command the agents to reset their positions to a random location within the search space after a predefined time period or when a large change is detected in the environment. The strategy was adapted for use in robotics by Yan et al. (2018) for leak source detection. In their proposed strategy, the average fitness value of each agent was calculated and a subset of agents that had a fitness value less than the system mean were instructed to restart their search process from a different location.

Preassigned Roles Besides dynamic response assignment, agents can be assigned a response pattern during their initialisation. Doing so allows for greater control over the system's behaviour and exploration and exploitation balance in exchange a lower level of system flexibility. In numerical optimisation to find a global optimum, Blackwell and Bentley (2002), Blackwell and Branke (2006), and Blackwell (2007) proposed

a PSO swarm inspired by the atomic model. This virtual swarm consists of agents clustered close to each other, which are dedicated to solution exploitation, and agents surrounding the central cluster, which are dedicated to exploration.

In the context of MRS, while agents with preassigned roles are more commonly used in environmental monitoring situations, such as the subCULTron MRS (Thenius et al., 2016). However, there have been some applications of this type of strategy in target tracking scenarios. In the tracking mobile targets, Wang and Cavallaro (2016) and Schranz and Andre (2018) proposed a hybrid camera system consisting of mobile and static cameras. In their systems, static cameras were used as dedicated exploratory agents to identify targets within the area of coverage. Mobile cameras were then used to exploit information coming from the static cameras and were assigned to pursue the detected targets as directed by a central controller.

2.4.2 Small Response Changes

In contrast to the large changes in state just discussed, another way of controlling the exploration-exploitation balance of a system is by permitting agents to effect smaller changes in their own response patterns. While these minor adjustments of an agent's behaviour on an individual level may not result in such drastic changes in a system's overall EED, they allow for more precise adjustments of the level of exploration and exploitation carried out by an MAS. A list of such strategies can be seen in Table 2.5.

Attraction-Repulsion Dynamics

Effecting changes to an agent's attraction and repulsion strength with respect to its neighbours or specified points in the environment is one of the more common small response changes used in both the development of strategies for use in MRS, as well as the modelling of natural systems (Kamimura, Masumoto, and Ohira, 2015; Wang et al., 2022). Increasing an agent's repulsion strength has the effect of augmenting the amount of exploration carried out by that agent. Conversely, increasing an agent's level of attraction favours more exploitative actions. Done across an entire MAS, this allows for regulation of a system's EED. When applied within an MRS context, the inter-agent repulsion also generates a collision avoidance behavior as a byproduct of this mechanism. Most repulsion vectors are calculated as follows:

$$V_i = \frac{a}{\|x_{i,t}\|^d} \frac{x_{i,t}}{\|x_{i,t}\|}, \quad (2.2)$$

where a is the set repulsion parameter, d is a constant, and $x_{i,j}$ is the vector from an agent i to a point of repulsion j . This point of repulsion is usually a neighbouring agent and using an equation similar that seen in Equation 2.2 results in higher repulsion strengths being generated when two agents are close together. Hence, when using such repulsion equations, agents are strongly repelled from each other when they are immediately next to each other (see Fig. 2.3), thereby discouraging excessive clustering and thus deterring over-exploitation of the environment.

Reference	Strategy	Task
Schumer and Steiglitz, 1968; Azad and Hasaengebi, 2014; Hansen and Ostermeier, 2001; Hansen, 2006	Adaptive step size	Optimisation
Blackwell and Bentley, 2002	Exponential inter-agent repulsion strength	Optimisation
Kernbach et al., 2009; Schmickl et al., 2009; Bodi et al., 2012; Bodi et al., 2015; Kengyel et al., 2016; Wahby et al., 2019	Varying agent wait time based on local fitness value	Optimal Area Aggregation
Pang et al., 2019; Nauta et al., 2020	Variable Lévy parameter	Resource Foraging
Sardinha, Dragone, and Vargas, 2020b	Variable Lévy parameter & interruption of Lévy walks	Resource Foraging
Hecker, Carmichael, and Moses, 2015	Random walk that gradually becomes more directed	Resource Foraging
Hayes, Martinoli, and Goodman, 2002; Jatmiko, Sekiyama, and Fukuda, 2007	Exponential inter-agent repulsion strength	Target Search
Sutantyo et al., 2013	Using Lévy walks and Firefly optimisation algorithm to generate points of attraction for each agent with different attraction weights	Target Search
Zhang et al., 2019a	SRLA and SRLAMR agent interaction schemes	Target Search
Coquet et al., 2019; Coquet, Arnold, and Bouvet, 2021	Exponential inter-agent repulsion and attraction strength	Target Tracking
Parker and Emmons, 1997; Parker, 2002; Kolling and Carpin, 2006; Kolling and Carpin, 2007	Linear inter-agent repulsion strength & variable target attraction strength	Target Tracking (CMOMMT)

TABLE 2.5: List of works using small response changes to influence the level of exploration and exploitation of an MAS.

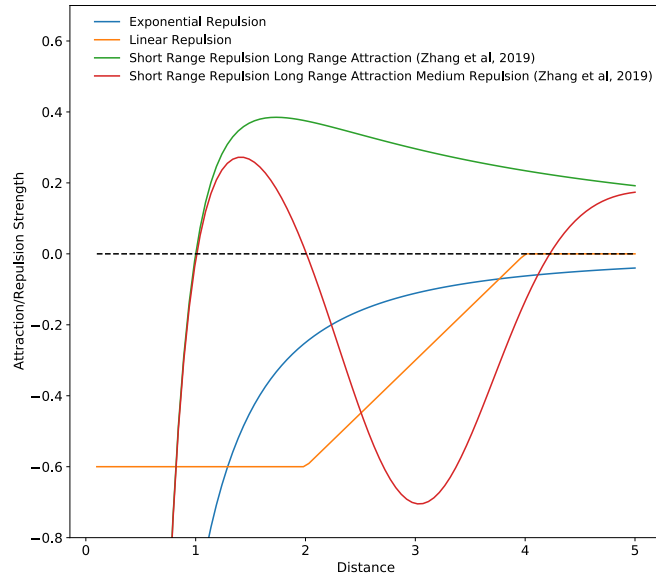


FIGURE 2.3: Inter-agent interaction forces generated by various repulsion and attraction schemes. Positive values indicate an attractive force and negative values indicate a repulsive force.

In a swarming MRS developed to find the source of a dynamic odour plume, Jatmiko, Sekiyama, and Fukuda (2007), inspired by the Charged PSO algorithm developed by Blackwell and Bentley (2002), increased the strength of an inter-agent repulsion parameter when robots came within a predefined distance of each other to avoid high levels of clustering within too small an area. It should be noted that in the experiments done by Jatmiko, Sekiyama, and Fukuda (2007), the size of the environment and the number of robots in operation, and hence the global average swarm density, are known *a priori*. This allowed for the authors to tailor the repulsion parameters and optimise the robot's search behavior accordingly (Hayes, Martinoli, and Goodman, 2002). A similar repulsion scheme was adapted by Coquet et al. (2019) and Coquet, Arnold, and Bouvet (2021) for use in their dynamic target tracking swarm.

In addition to inter-agent repulsion to prevent excessive aggregation when tracking multiple mobile targets, Parker and Emmons (1997) also varied an agent's attraction strength to a target based on the distance between an agent and the target, with targets further away triggering a weaker attraction response. Furthermore, the authors also implemented agent-target repulsion when an agent moved too close to a target to prevent excessive exploitation. Developing this strategy further, Parker (2002), Kolling and Carpin (2006), and Kolling and Carpin (2007) reduced the weight of the point of attraction generated by targets already being tracked, preventing the overlap of robot sensing areas, further reducing the over-exploitation of a target. In these works, a linear repulsion function was used, as seen in Fig. 2.3, instead of the exponential function that is normally used.

In a target search task, Sutantyo et al. (2013) ignored inter-agent repulsion and focused only on the use of attraction by combining the Lévy walk together with a Firefly Optimisation algorithm to generate a velocity vector for each agent at each time-step. An agent's velocity is influenced by the attractiveness value of other agents within its communications range, encouraging exploitation, and a random point within the environment obtained from a Lévy distribution, encouraging exploration. Using the proposed strategy, an agent's attractiveness is based on a function on the amount of time elapsed since it last encountered a target. An agent that sees a neighbour with a high level of attractiveness will bias its movements more towards that neighbour rather than towards the point of exploration.

Besides the inter-agent repulsion behaviour more commonly utilised, Zhang et al. (2019a) also implemented an inter-agent short range repulsion and long range attraction (SRLA) component within their strategy based on inter-agent distances, effectively preventing excessive exploration for their k -capture game carried out by an MAS. In addition to this, they developed a Short Range Repulsion Long Range Attraction Medium Range Repulsion (SRLAMR) scheme where the inter-agent force transitioned between repulsion and attraction depending on the inter-agent distance. These two attraction and repulsion schemes were derived from the work done by Gazi and Passino (2004) and Chen, Liao, and Chu (2011) and can be seen in Fig. 2.3. Comparing these schemes with the exponential repulsion schemes, the authors found that using the exponential repulsion scheme resulted in too much exploration, preventing targets from being captured, while the SRLA and SRLAMR schemes promoted agent clustering, facilitating the capture of targets. However, the SRLAMR scheme outperformed the SRLA scheme due to the former resulting in the dynamic formation and dissolution of agent clusters over time. These clusters were also distributed across the search space, demonstrating a balance between exploration and exploitation, allowing the system to capture fast-moving targets. By combining and adjusting the strength of separate inter-agent attraction and repulsion fields, Coquet, Arnold, and Bouvet (2021) have demonstrated that a stable equilibrium position can be attained where agents maintain a fixed relative position to each other even though the entire swarm may be in motion. This allowed for the overall surface of an MRS, and hence the overall EED of the swarm, to be controlled even though the entire system may be moving.

Random Walks

As seen in Sections 2.4.1 and 2.4.1, a common exploration strategy for MAS in environments where there is a lack of information is the use of random walks. In such environments, policies based on attraction-repulsion dynamics may not be viable due to the lack of stimuli to drive agent activity. To operate in such information sparse environments, random walks, such as Lévy walks, may be used as a method to promote area exploration. However, the use of Lévy walks may result in inefficient searching due to possible inter-robot collisions and coincidental clustering of robots. As such, Pang et al. (2019) developed a random walk in which agents determine their walk sizes based on a local swarm density estimation. This allows a robot to carry out longer random walks when in less crowded areas, thereby promoting exploration. Conversely, their strategy

also favours shorter walks when a robot is located of higher agent densities, thereby promoting exploitation.

Applications of an adaptive step size are also commonly found in computational optimisation algorithms and usually take into account a candidate solution's direction of travel (Schumer and Steiglitz, 1968; Azad and Hasançebi, 2014; Hansen and Ostermeier, 2001; Hansen, 2006). In such strategies, the step size of a candidate solution is varied based on the path taken by the solution; the step size is decreased should the solution stay around the same region of the search space for several iterations, enabling better exploitation, and increased should the solution 'move' in the same direction for several consecutive time-steps, facilitating exploration.

In a similar fashion, to carry out their resource foraging task, Nauta et al. (2020) also varied the Lévy parameter, μ , of individual agents using a parameter based on the amount of time elapsed since an agent's last target encounter. Using this strategy, an agent's Lévy parameter value continuously decreases towards unity while an agent does not encounter any targets. Doing so allowed for an agent to gradually prioritise exploration if it does not encounter a resource block for long periods of time. Similar to this, Sardinha, Dragone, and Vargas (2020b) also explored varying a parameter that determines an agent's propensity of prematurely interrupting a Lévy walk should the agent experience a decreasing number of resource encounters. By increasing the chances of interruption, the agent is more likely to change its search path and explore the search area.

While carrying out a foraging task, Hecker, Carmichael, and Moses (2015) used an informed random walk that is initially un-directed and localised to look for patches of resources, permitting a higher level of exploration. As an agent gathers information about resource locations, these walks become less random and more directed as agents tend to move directly towards higher yielding patches. This moves the system's exploration-exploitation balance in favour of exploitation, allowing agents to exploit high rewarding resource patches while retaining some randomness in the walk permits them to explore the environment for more profitable locations.

Time Intervals

The use of environment sampling intervals as well as communication time intervals is also a method that can be used to control the level of exploration and exploitation carried out by an MAS. This is the primary method used by the BEECLUST algorithm first proposed by Kernbach et al. (2009), Schmickl et al. (2009), and Bodi et al. (2012) that allowed a swarm to find and aggregate in an area with high light intensity. This algorithm was formulated after observing bees tending to cluster in regions of higher temperature and in regions where clusters are already present. When implemented in an MRS, agents attempted to find an area illuminated by the source with the highest light intensity where the light intensity of the sources varied over time. Using this algorithm, agents measure their local environment's light intensity only when they encounter other agents and proceed to wait in place for a period of time. This waiting time is determined by each agent individually and increases with the measured light intensity. This leads to a positive feedback loop where longer wait times lead to

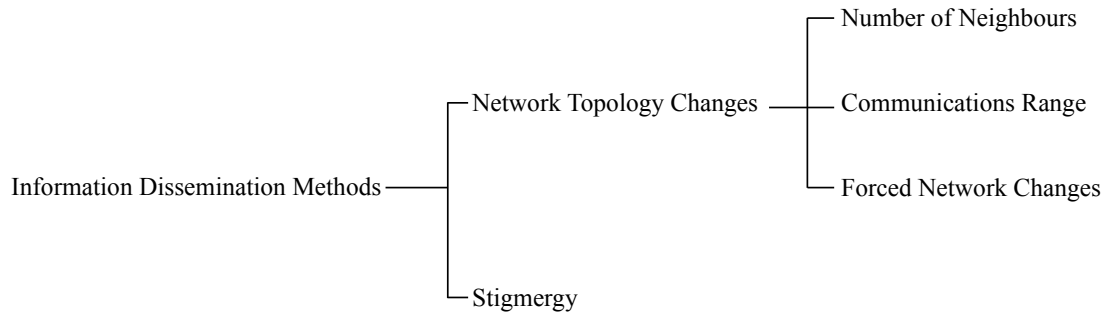


FIGURE 2.4: Classification of different information dissemination methods to control an MAS' exploration-exploitation balance.

more aggregation in areas with higher light intensity. This phenomenon is known as Motility-Induced Phase Separation (MIPS) (Cates and Tailleur, 2015) and eventually results in higher levels of exploitation carried out by the system in areas of high agent density. Algorithms making use of MIPS were later studied by Bodi et al. (2015) for use in underwater environments and demonstrated in physical robot platforms by Kengyel et al. (2016).

While effective, the positive feedback loop caused by MIPS may cause the system to over-exploit a specific area and result in lower levels of responsiveness in the presence of a highly dynamic light source. As such, Wahby et al. (2019) modified algorithm to account for an agent's local swarm density when determining its wait time. This modification allowed an agent to move off earlier when located in an area of high swarm density, preventing over-exploitation and improving the system's ability to respond and adapt to a highly dynamic environment.

2.5 Information Dissemination Methods

The second broad category of approaches towards controlling the exploration and exploitation dynamics of an MAS is based on adapting the way information is disseminated across the system. This system-wide information transfer can be achieved by means of direct or indirect communication strategies. Indirect methods are often observed in nature with agents altering the environment to convey information to fellow agents: this process is known as stigmergy. For example, some ant colonies use pheromones to guide worker ants to various food sources (Sumpter, 2010; Perna et al., 2012), which serves as the inspiration for various optimisation algorithms, such as the Ant Colony Optimisation algorithm (Dorigo, 1992). With engineered systems, radio transmissions offer a direct means of communication and is the most common among MRS for dissemination of information. These direct methods are generally used in tandem with an agent response method that controls the amount of information an agent has access to. By doing so, the actions and responses of neighbouring agents can be influenced indirectly and can have large effects on the system's collective response to the environment. A visual classification of the types of information dissemination methods used to control a system's EED can be found in Fig. 2.1 and a list of these strategies can be found in Table 2.6.

Reference	Strategy	Task
Mateo, Kuan, and Bouffanais, 2017	Changing interaction radius	Distributed Consensus
Mateo et al., 2019	Changing number of communication neighbours	Leader-Follower Task
Janson and Middendorf, 2005; De Oca et al., 2009; Liu et al., 2016; Blackwell and Kennedy, 2018	Changing number of communication neighbours	Optimisation
Tillett et al., 2005	Dynamic sub-swarm membership	Optimisation
Talamali et al., 2021	Adjusting communications range	Optimal Size Aggregation
Tang et al., 2020b	Changing an agent's number of neighbours in network based on fitness value	Target Search
Couceiro, Rocha, and Ferreira, 2011c; Couceiro, Rocha, and Ferreira, 2011a; Couceiro et al., 2014b; Couceiro and Portugal, 2018	Dynamic sub-swarm membership	Target Search
Esterle and Lewis, 2017	Varying communication link strength and number of communication neighbours	Target Tracking (CMOMMT)

TABLE 2.6: List of works using different information dissemination strategies to influence the level of exploration and exploitation of an MAS or MRS.

2.5.1 Changes in Network Topology

MAS that use direct communication methods can control the information flow within the system by varying several possible communication network parameters, such as number of neighbours (for individual agents), agent's communication range, and various techniques like sub-swarm assignment and forced network change. These are further elaborated in the following sub-sections.

Number of Neighbours

One of the network parameters that can affect the speed at which information is spread around the network is the number of direct (or immediate) neighbours that each agent possesses. It is worth stressing that the concept of 'neighbour' shall not be limited to classical definition according to a metric or Euclidean distance. Instead, this concept should be considered in its most general form, which is conveniently captured by the network theoretic concept of node degree (Shang and Bouffanais, 2014). In general, most studies consider the number of neighbours as a system-level parameter—i.e., all the agents i have the same number of neighbours $k_i = k$ —although it is absolutely possible to envisage a heterogeneous system, in which the k_i values are different. When considering this parameter k , it is crucial to distinguish between directed and un-directed networks. Specifically, with a directed network, the information flows across the system in one direction, i.e., if agent A is a neighbour of agent B then the information flows from A to B , but not necessarily the other way round. Directed networks make the relationship between agents asymmetric in terms of information flow. On the other hand, with un-directed networks, that relationship is symmetric, meaning that the information flows in both directions when two agents are connected.

Through these repeated local information exchanges, an agent is able to form local decisions based on the information gathered by itself and its immediate neighbours—still in the network sense. For instance, using the PSO algorithm, changing the number of neighbours an agent communicates with is one of the key methods for controlling the level of exploration and exploitation carried out by the system (Janson and Middendorf, 2005; De Oca et al., 2009; Liu et al., 2016; Blackwell and Kennedy, 2018). By increasing k , information about the global 'best' solution can be propagated more swiftly around the system, thereby encouraging exploitation. Conversely, by reducing the degree k , information propagation is slowed down. This prevents agents from over-exploiting the same piece of information, pushing the EED balance slightly in favour of exploration.

To prevent agents from being trapped in local optima, Tang et al. (2020b) developed a robotic PSO strategy using a synthesised dynamic neighbourhood. In this framework, an agent expands its neighbourhood should it be determined to be trapped in a local optimum. By using such expanding neighbourhoods, trapped agents have a chance to find and follow a 'leader' with a better fitness value, thereby progressing towards the global optimum. This concept was first introduced by Mateo, Kuan, and Bouffanais (2017) and Mateo et al. (2019) in distributed consensus tasks and will be explored in greater detail later in this thesis as it forms a key component of the proposed target search and track strategy.

Communication Range

For the operation of MRS, the maximum communications range between two robots is of significant interest as it is one of the key factors that determines the maximum number of neighbours in an agent's communications network. As such, many studies carefully focus their investigations on the limitations associated with the communication range.

Similar to increasing the size of an agent's neighbourhood, expanding an agent's communication range also serves to increase the number of possible information exchanges that an agent can carry out. As demonstrated by Talamali et al. (2021), smaller communication range, and hence less information exchange, results in agents being more willing to adapt to their local environments. However, this over-willingness to adapt leaves an agent vulnerable to random fluctuations of its neighbours' output. Along the same vein, Coquet et al. (2019) also showed that increasing the communication range made swarming systems less sensitive to noise and better able to track multiple moving targets. This occurs because increasing an agent's communications range allows for information to be disseminated more quickly around the system. Points of interest picked up by a single agent would be more attractive compared to local optima occurring due to environmental noise. This higher attractiveness will cause agents to shift their behaviours in favour of exploitation and converge quicker on that point of interest.

Despite the fast rate of convergence, care needs to be taken when setting an agent's communications range, as previously mentioned. As shown by Wang et al. (2021), while increasing communications ranges may initially reduce the amount of time required for the system to converge, increasing it past a certain distance may no longer be of any benefit to the swarm and may be to the detriment of the system. In addition to the lower level of responsiveness to the environment, as mentioned by Talamali et al. (2021), higher communications ranges may require the use of more powerful radio transmitters, increasing the cost, power consumption, and weight of an individual swarming agent.

Forced Network Changes

Should a fixed level of connectivity be desired, changes can be made to the network's structure by changing an agent's communications neighbours. This allows an agent to maintain the same communication range or number of neighbours, while allowing for more information diversity in the agent's network. Esterle and Lewis (2017) and Esterle and Lewis (2020) dealt with online multi-object k -coverage. The strategy is inspired by an ant foraging method that consists in both negative and positive feedback loops. When two cameras are connected to each other, this connection is considered as a link and that will be reinforced as long as the cameras continue to 'see' the same object. The strategy also allows for the weakening of this connection as all link strengths decay over time. This decay opens the possibility for it to be used in dynamic environments as the system can unlearn the links, preventing the over-exploitation of outdated information sources.

In the same two works, various network structures are used, namely the k -nearest, k -furthest, and k -random networks. The k -nearest and k -furthest networks are based on Euclidean distance, while the k -random network is based on a set of randomly chosen neighbours across the entire system, allowing the communication to gradually expand to “all” cameras. Using topologies such as the k -furthest or k -random networks essentially allowed the agents to explore a larger area by permitting the gathering of information from neighbouring agents further away. This increased amount of exploration was studied while using various response models (e.g., moving to assist in the tracking of the closest target, moving to assist in the tracking of the target with the lowest number of pursuing agents, etc.). Using wider networks paired with the response model increased the level of exploration of an agent as well as allowed the agents to select the most appropriate source of information to exploit that will benefit the system the most.

As mentioned previously, systems employing strategies based on the Darwinian PSO (Tillett et al., 2005) and the derivative RDPSO algorithms (Couceiro, Rocha, and Ferreira, 2011c; Couceiro, Rocha, and Ferreira, 2011b; Couceiro et al., 2014b; Couceiro and Portugal, 2018) also make use of changing network structures. By allowing the system to form dynamic sub-swarms through the constant inclusion and exclusion of agents, the system is able to simultaneously find and exploit different sources of information.

2.5.2 Stigmergy

Should an MAS operate in an environment where direct agent-to-agent communications are denied, indirect methods, such as stigmergy, can be used. In this framework, an agent is able to elicit a response from other agents in the system by leaving traces in the environment (Theraulaz and Bonabeau, 1999). This concept was first introduced in Grassé (1959) and has been applied in various MAS. A list of these strategies can be found in Table 2.7.

While the most popular usage of stigmergy is the Ant Colony Optimisation algorithm, first proposed by Dorigo (1992), it has also seen some applications in target search MRS. While using robots to carry out a target search task, (Herianto, Sakakibara, and Kurabayashi, 2007; Herianto and Kurabayashi, 2009; Johansson and Saffiotti, 2009) spread out RFID data carrier tags around the physical operating space to act as pheromones within the environment. A similar strategy was also employed by (Khaliq, Di Rocco, and Saffiotti, 2014; Khaliq, 2018), who used a physical RFID grid to generate a map exploitable by a swarm of robots, guiding them towards a target. Using this type of strategy, robots are equipped with an RFID writer/reader and update the pheromone levels of the tags when they are encountered. Doing so allows an agent to exploit the environmental information by following a gradient descent algorithm towards the target. Similar to their counterparts in computational optimisation, these pheromones are also subject to evaporation, preventing the robots from exploiting outdated information. Tang et al. (2019) and Tang et al. (2020a) later built on this work by supplementing the strength of the pheromone with a vector pointing the direction of areas with higher pheromone concentrations. This allows an agent to carry out more effective (targeted) exploitative actions, improving search efficiency and success rates when searching for both dynamic and static targets.

Reference	Strategy	Task
Dorigo, 1992	Ant Colony Optimisation with pheromone evaporation	Optimisation
Herianto, Sakakibara, and Kurabayashi, 2007; Herianto and Kurabayashi, 2009; Johansson and Saffiotti, 2009; Khaliq, Di Rocco, and Saffiotti, 2014; Khaliq, 2018	Gradient decent with physical data carriers	Target Search
Tang et al., 2019; Tang et al., 2020a	Gradient decent with vectorial pheromones and physical data carriers	Target Search

TABLE 2.7: List of works using stigmergy to influence the level of exploration and exploitation carried out by an MAS.

2.6 Summary

The exploration–exploitation dilemma is a common challenge faced by many systems, including those encountered by various multi-agent and multi-robot systems when searching for and tracking dynamic targets. This dilemma stems from the fact that, in general, exploration and exploitation tend to be mutually exclusive tasks.

The allocation of agents to these exploration and exploitation tasks, be it via automated or human-driven means, is a nontrivial task; a large proportion of agents carrying out exploration would indeed result in many points of interest being identified, however, the system would not be able to fully benefit from these points. Conversely, while a large proportion of agents carrying out exploitation would increase the rate at which the system profits from the information currently available, the system would not be able to display enough flexibility to operate effectively in a dynamic environment—in particular a rapidly evolving one—and may not be able to maximise the amount it profits from the environment.

Given the challenging nature of this problem and the diverse range of applications for MAS and MRS systems, there is a large body of literature dedicated to studying the balance between exploration and exploitation in such systems. These studies and works often originate from different disciplines, (e.g., autonomous multi-agent dynamics, field robotics, and service robotics) and utilise various tools and methods to develop solutions and strategies in relation with their specific problems (e.g., control theory, simulations of self-organised behaviors, and physical experimentation). Such a wide variety of missions would inevitably result in different optimum exploration-exploitation balances for different systems. As such, any developed swarm strategy must either be tailor-made for a specific task or risk sub-optimal performances, or be

able to adapt the system's overall behaviour to suit a range of operating conditions (Kit, 2021).

Given how applicable the problem of balancing exploration and exploitation is in many different domains and disciplines, this chapter focused on identifying strategies commonly used in target search and tracking scenarios and classifies the similarities found in these solutions and frameworks. The solutions used to control system exploration and exploitation fall into two broad categories, namely "agent response methods" and "information dissemination methods." The latter can be further broken down into "network topology methods", where a system's communications network parameters are changed to regulate the flow of information across the MAS, and stigmergy, where an agent is able to indirectly elicit a response from other agents in the system by leaving behind a signal in the environment. Similarly, the broad category of "agent response methods" can be further broken down into: (i) methods that cause an agent to display large and drastic changes in the way it deals with the environment, (ii) methods that result in smaller changes in an agent's response to its environment, and (iii) the assignment of agents to areas of interest or tasks found in the environment.

In addition, a summary of the methods used in various studies to quantify the level of exploration and exploitation carried out by an MAS was also presented in this chapter. Such quantitative analyses rest upon the proper definition of metrics, which are key to estimating an MAS' exploration-exploitation balance, thereby allowing one to see if a proposed strategy is working as intended, and make adjustments as needed. However, as is apparent in this review of the literature, metrics tend to be closely tied to their specific applications/problems and there is no generalised metric available to measure exploration and exploitation. This important point has also been raised in previous reviews, and a similar observation was also made in reviews about swarm robotics (Brambilla et al., 2013) as well as human-swarm interaction (Kolling et al., 2016). The lack of a single/absolute metric applicable to different problems limits one's ability to truly rank different approaches and strategies. Even if there is no guarantee that such a universal metric exists, the community working on these problems would benefit from ways to compare and rank current MAS strategies.

Chapter 3

Non-Evasive Targets

3.1 Introduction

The problem of searching for and tracking multiple targets is a well explored area of research. In this field, the leading strategies for dynamic target search and tracking are normally based around centrally controlled agents using a predetermined search pattern to perform a systematic sweep of a given search area (Meghjani, Manjanna, and Dudek, 2016; Medina et al., 2017). However, these search methods are rendered ineffective when searching for and tracking a target moving faster than the individual searchers.

While methods involving the use of swarming multi-robot systems (MRS) have been proposed, majority of the current literature tends to focus on cases where the targets move slower than or as fast as their pursuing agents (Senanayake et al., 2016; Khan, Rinner, and Cavallaro, 2018). This is especially the case when evasive targets are being tracked because of the long-standing assumption that in these situations the targets will always be able to evade their pursuers. These assumptions lead to a false sense that the problem of tracking a fast moving target is impossible for the pursuing swarm (Parker and Emmons, 1997; Parker, 2002), even though it has been shown that it is possible when a swarm has vision over the entire operating environment (Janosov et al., 2017; Zhang et al., 2019a).

In addition to this, the use of swarming MRS gives the end-user the opportunity to incrementally upgrade the robotic system, invariably leading to the use of heterogeneous

The content in this chapter has been published in the following papers:

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)*, 672–680. DOI: 10.5555/3398761.3398842

Kwa, H.L., Tokić, G., Bouffanais, R., and Yue, D.K.P. (2020). "Heterogeneous Swarms for Maritime Dynamic Target Search and Tracking". In: *Global OCEANS 2020: Singapore-U.S Gulf Coast*, DOI: 10.1109/IEEECONF38699.2020.9389145

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*. DOI: 10.1162/isal_a_00376

swarming MRS. However, such systems have not been well studied and incorporating upgraded agents into a swarm may result in degraded mission performances (Altshuler, 2009; Vallegre et al., 2018). As such, the improved abilities of upgraded agents need to be taken into account in the design of the system to benefit from the upgraded agents' capabilities.

In this chapter, a fully decentralised swarming strategy that is capable of adjusting its level of engagement with a fast-moving target, and hence its exploration and exploitation dynamics (EED). The base algorithm was inspired by the Particle Swarm Optimisation (PSO) algorithm and was augmented through the utilisation of an adaptive repulsion behaviour, as well changing the level of inter-agent connectivity in an inter-connecting k -nearest neighbour communications network. By varying the topological distance between agents, i.e., the degree k of the k -nearest neighbour network, one can change the system's balance between exploitative and exploratory behaviours. As such, agents can be made to prioritise the conduct of either exploratory or exploitative actions. In doing so, the system autonomously adapts its collective dynamics to maximise its tracking performance while pursuing targets of different speeds and quantities.

This strategy is studied through its implementation in both homogeneous and heterogeneous swarms in simulations. Through these simulations, an optimal collective performance of the swarm corresponding to a specific connectivity, k^* , is found. This swarm strategy for dynamic search and tracking is experimentally tested and validated in the homogeneous case with a decentralised swarm of eight wheeled miniature robots operating in the absence of any supporting infrastructure.

3.2 Related Work

3.2.1 Particle Swarm Optimisation

The Particle Swarm Optimisation algorithm was originally introduced by Kennedy and Eberhart (1995) as a means to find the global optima of stationary multidimensional and nonlinear functions. Recently, the algorithm was brought into the realm of robotics to achieve effective target searches with swarming MRS (Jatmiko, Sekiyama, and Fukuda, 2007; Couceiro et al., 2014a; Yan et al., 2018; Coquet et al., 2019), with the majority of applications being in static environments. Several modifications enabling this translation have been made, including the addition of obstacle or inter-agent collision avoidance behaviours (Couceiro, Rocha, and Ferreira, 2011b), as well as the implementation of a limited communications range (Pugh and Martinoli, 2007; Coquet et al., 2019; Yang, Wang, and Bauer, 2019). The fact that PSO is a metaheuristic makes it a highly compelling approach for decentralised MRS operations in dynamic search-spaces.

There are two main challenges of implementing a PSO-based target search and tracking strategy in dynamic environments that have not been sufficiently addressed. These challenges are: (1) solving the problems caused by outdated memory, and (2) excessive exploitation of the search environment at the expense of exploration (Jordehi, 2014). In both cases, the swarm converges on the target but is unable to continue tracking the target's movements due to insufficient exploratory actions performed by the agents

once engaged with the target. This highlights the requirement that the swarm needs to be able to constantly explore the search space, even after the target is found.

To promote sufficient exploration of the search space, swarm reinitialisation procedures have been considered. This involves randomising the particle positions and fully restarting the entire PSO process when the ‘global best’ ($gBest$) value has not been updated in several iterations (Eberhart and Shi, 2001; Liu et al., 2017). However, methods using such reinitialisation techniques can only be considered when PSO is used for computational optimisation. They are neither practical nor feasible in the physical world and as such, Jatmiko, Sekiyama, and Fukuda (2007) have proposed dispersing the robotic swarm if the $gBest$ value has not changed after a specified number of time-steps. Another strategy to promote exploration of the search environment is to employ agents with a constant repulsive parameter, preventing them from clustering (Blackwell and Bentley, 2002). However, such approaches require an all-to-all communications network to facilitate inter-agent repulsion. This makes MRS using such methods essentially centralised and less tolerant of failures in the communications network.

3.2.2 Network Connectivity

Another key factor to take into account when devising PSO-based search and tracking strategies is the connectivity of the agents involved when performing such tasks. For the application of PSO in the optimisation of non-linear functions, the most commonly studied network structures are the static $lBest$ (ring) and $gBest$ (all-to-all) network topologies, with the $gBest$ topology being the most popular network utilised (Liu et al., 2016). Typically, network connections are determined during the initialisation of the particles and do not change while the PSO process is underway, leading to static networks. It has recently been recognised that agents in PSO networks with lower amounts of connectivity (i.e. low degree) perform more exploratory actions achieving higher global search performance. In contrast, agents in networks with greater connectivity (i.e. high degree) perform more exploitative actions and more suited to local search tasks (Blackwell and Kennedy, 2018). It is worth pointing out that the network topology plays a central role in the effectiveness of distributed information exchanges underpinning the collective response of a swarm (Rosenthal et al., 2015).

While the use of static networks does not pose any problems with the PSO algorithm’s applications *in silico*, using such methods *in robotico* with an MRS operating within an unstructured dynamic environment poses significant challenges. This includes the limited communication ranges and limited information processing capabilities of a robot, resulting in constraints on the number of connections each robot can establish and sustain. As such, dynamic PSO networks based on metric-distance neighbours have been proposed and studied (Pugh and Martinoli, 2007; Yang, Wang, and Bauer, 2019). In the study of these networks, it has been shown that larger communication ranges result in faster convergence times for the swarm (Pugh and Martinoli, 2007). This holds true until the communications radius of each member is able to cover the entire swarm, at which point, an all-to-all network is formed and there is no further reduction in convergence time (Yang, Wang, and Bauer, 2019). These faster convergence times are due

to the better local search properties of communications networks with higher degrees of connectivity.

Recently, it has also been shown that regardless of the technical limitations of the individual robotic units, there exists an optimal level of connectivity between swarming agents involved in a decentralised decision-making process while tracking a dynamic signal. Interestingly, this optimal connectivity is directly related to speed of the driving signal (Mateo et al., 2019). This presence of an optimal level of connectivity that maximises a system's performance has also been observed in static environments that yield very little information to the agents (Crosscombe and Lawry, 2021; Liu, Crosscombe, and Lawry, 2021; Kwa and Bouffanais, 2022). Furthermore, it has also been shown that an excessive amount of interaction limits a swarm's ability to collectively respond to perturbations (Mateo, Kuan, and Bouffanais, 2017). This implies that while tracking a dynamic target, instead of relying on all agents within a set communication range, tuning the level of connectivity of a swarm may result in a better collective tracking performance. There have been efforts made to implement dynamic network topologies in robotic swarms, mainly through adjusting the communications ranges of individual agents. While efforts have been made to implement dynamic network topologies in robotic swarms, this effect of varying the level of connectivity of a swarm operating in a dynamic environment has largely been ignored.

3.2.3 Heterogeneous Agents

As mentioned previously, the use of swarming MRS brings with it the advantages of system modularity. This often overlooked benefit allows for agents with different properties and behaviours to be implemented within the same swarm, resulting in heterogeneous swarms. Heterogeneous swarm compositions can be utilised in several ways including employing agents with different communication ranges to reduce the overall system cost (Christensen et al., 2015) and improving a system's mapping capabilities by partitioning a search-space and assigning the partitioned regions to robots according to their movement and sensing capabilities (Shi et al., 2020). Strickland et al. (2018) also demonstrated that through optimising the composition of agents with different capabilities, the total amount of energy expended during a site defence task can be minimised. However, as shown by Esterle (2018b), this optimum composition also depends on the demands of the assigned task. Behavioural heterogeneity has also been shown to be important when operating in evolving environments. For example, in their social learning task, Lawry (2022) demonstrated that a behaviourally heterogeneous system is critical to operating within a dynamic environment, allowing the agents to explore the environment and exploit the gathered information at the same time.

In addition to the aforementioned benefits, a system's modularity presents the opportunity to incrementally introduce upgrades to a small proportion of agents without having to apply the upgrade to the entire swarm. This allows for the testing of agent upgrades within their actual operating environment before implementing the change across the rest of the swarm. However, such incremental upgrades made to a system will necessitate the use of heterogeneous swarms. It has been shown that the simple implementation of upgraded agents in a swarming system does not necessarily translate into an improvement of system performances. Altshuler (2009) showed that even if

a homogeneous swarm of agents with increased communication ranges were deployed to carry out a target search task, these improved capabilities need to be taken into account before any improvements to the system's performance were observed. This result suggests that the improved abilities of upgraded agents may need to be taken into account in the design of the system to benefit from the new heterogeneous nature of the swarm.

3.3 Search and Track Strategy

The strategy used to enable a swarm to search for and track a fast-moving target consisted of two behavioural patterns: (1) promotion of agent aggregation around a common location of interest, serving as a driver of exploitation, and (2) an adaptive inter-agent repulsion behaviour, serving as a driver of exploration. The aggregation and repulsion behaviours, which were inspired by the Charged Particle Swarm Optimization algorithm (CPSO) (Blackwell and Bentley, 2002) and the Social-Only PSO (Engelbrecht, 2010), generated two velocity vectors at each time-step. These two components were summed to calculate an agent's its final velocity vector:

$$\mathbf{v}_i[t] = \mathbf{v}_{i,\text{att}}[t] + \mathbf{v}_{i,\text{rep}}[t], \quad (3.1)$$

where $\mathbf{v}_{i,\text{att}}[t]$ is the velocity vector generated by the attractive component and $\mathbf{v}_{i,\text{rep}}[t]$ is the velocity vector generated by the adaptive inter-agent repulsion component. The level of social interaction was controlled by varying the value of a parameter, k , of an inter-connecting k -nearest neighbour communications network that allowed for the sharing of target information among agents in the swarm.

The overall strategy employed in the system is detailed in Algorithm 1. The individual components of this algorithm will be explained in the following sections.

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  $i \in [1, N]$  do
4:     Set  $\mathbf{p}$  using Algorithm 2
5:     Calculate Attraction Component  $\mathbf{v}_{\text{att},i}$  with Eq. 3.2
6:     Calculate Repulsive Component
7:      $\mathbf{v}_i[t] \leftarrow \mathbf{v}_{\text{att},i}[t] + \mathbf{v}_{\text{rep},i}[t]$ 
8:      $\mathbf{v}_i[t] \leftarrow \mathbf{v}_i[t] / v_{\text{max}}$ 
9:      $\mathbf{x}_i[t + 1] \leftarrow \mathbf{x}_i[t] + \mathbf{v}_i[t]$ 
10:   end for
11:    $t \leftarrow t + 1$ 
12: end while

```

3.3.1 Swarm Connectivity

The interaction network among swarming agents plays a key role in the effectiveness of the proposed strategy. In particular, the level of connectivity—i.e. the degree—can be adjusted with significant effects on the collective dynamics (Bouffanais, 2016). As previously mentioned, higher levels of connectivity in computational PSO networks lead to higher levels of local exploitation, and consequently a higher degree of aggregation within the search space, while a low degree causes the swarm to prioritise exploration of the environment (Blackwell and Kennedy, 2018). Changing a swarm’s level of connectivity has also been shown to have considerable effects on the collective dynamics and level of responsiveness of a swarm (Mateo, Kuan, and Bouffanais, 2017; Mateo et al., 2019; Horsevad et al., 2022).

It has been shown that in flocks of starlings, birds interact with only 7 to 8 nearest neighbours on average, regardless of the distance between them (Ballerini et al., 2008). Such a topological neighbourhood concept can be represented by a k -nearest neighbor graph. Here, the value of the degree k , which is identical for all agents, is used to characterise the level of connectivity within the swarm. At each time-step, any agent i identifies its k -nearest topological neighbours and sets its point of attraction, $\mathbf{p}[t]$, to the observed target location as measured by itself or its neighbours. At this point, $\mathcal{N}_i[t]$ is defined as the set of k indices corresponding to the topological neighbours of agent i at time-step t . This is in contrast to the standard PSO algorithms and aforementioned variants that largely rely on static communications networks, where agents are assigned neighbours during the system’s initialisation that do not change over the course of the task. This is also in contrast to the commonly utilised metric distance neighbourhood where all agents within a given radius are considered as neighbours (Coquet et al., 2019; Coquet, Arnold, and Bouvet, 2021).

3.3.2 Attraction and Aggregation

In the swarm, an attractive component is used to encourage agents to aggregate at a point of interest, $\mathbf{p}[t]$. 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 neighbours and attempt to track targets detected by its neighbours. At this point, it is important to note that in the framework used, the neighbourhood of an agent is to be understood in the network sense. As such, an agent i has as many neighbours as its degree k . Also, since time-varying network topologies are considered, it should be noted the neighbourhood of each agent evolves over the course of the task duration. Given this dynamic network topology, all agents set their own $\mathbf{p}[t]$ using Algorithm 4.

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:

$$\mathbf{v}_{i,\text{att}}[t + 1] = \omega \mathbf{v}_i[t] + cr(\mathbf{p}[t + 1] - \mathbf{x}_i[t + 1]). \quad (3.2)$$

Algorithm 2 : Point of Attraction Update Algorithm for Non-Evasive Targets

```

if Agent detects target then
   $\mathbf{p}_i[t] \leftarrow$  Target's position
else
  Determine  $\mathcal{N}_i = \{j \in [1, N] \text{ s.t. Agent } j \text{ is a } k\text{-nearest neighbour of Agent } i\}$ 
  if  $\exists j \in \mathcal{N}_i \text{ s.t. } \mathbf{p}_j[t] = 1$  then
     $\mathbf{p}_i[t] \leftarrow \mathbf{p}_j[t]$ 
  else
     $\mathbf{p}_i[t] \leftarrow \emptyset$ 
  end if
end if

```

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 case where multiple targets are present, these targets will never overlap each other and agents do not assign unique identifiers to tracked targets. It should be noted that in the proposed strategy, unlike the social-only PSO model that uses an infinite memory length, memory is not used by the system. This is done to prevent agents from exploiting outdated target positional information.

3.3.3 Adaptive Repulsion

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

The inter-agent repulsion behaviour adopted is based on the one used in the BoB environmental monitoring swarm developed by Valleggra et al. (2018) and Zoss et al. (2018). Using this behaviour, an Agent i with topological neighbours j calculates its individual repulsion velocity as follows:

$$\mathbf{v}_{\text{rep},i}[t] = - \sum_{j \in \mathcal{N}_i} \left(\frac{a_R[t]}{r_{ij}[t]} \right)^d \frac{\mathbf{r}_{ij}[t]}{r_{ij}[t]}, \quad (3.3)$$

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-neighbour distance in equilibrium configuration (Valleggra et al., 2018; Coquet, Arnold, and Bouvet,

2021). As such, these two parameters were set so that the swarm was able to cover the entire search area when there was no target found.

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 concept of an agent's tracking state, $S_i[t]$, is introduced. When an agent has information of a target's location, either through local measurements or communications with its direct neighbours, it is assigned a tracking state of 1, moves towards the target location, and reduces its $a_R[t]$ value until a minimum value is reached. When an agent has no target information, it enters an exploratory state, assigns itself a tracking state of 0, and increases its $a_R[t]$ value until a maximum value is attained. Formally, the tracking state is assigned by considering $S_i(\mathbf{x}_i[t], t) = 0$ if $\mathbf{p}_i[t] = \emptyset$, and 1 otherwise. The adaptive repulsion behaviour used to obtain the repulsion component is summarised in Algorithm 3.

Algorithm 3 Adaptive Repulsion

```

Initialise  $a_{R,\min}$ ,  $a_{R,\max}$ ,  $d = 6$ ,  $\delta_{\text{explore}} = 0.1$ , and  $\delta_{\text{track}} = 0.75$ 
while System active do
  if  $S_i[t] = 1$  then
    if  $a_R > a_{R,\min}$  then
       $a_R \leftarrow a_R - \delta_{\text{track}}$ 
    end if
  else
    if  $a_R < a_{R,\max}$  then
       $a_R \leftarrow a_R + \delta_{\text{explore}}$ 
    end if
  end if
  Calculate  $\mathbf{v}_{\text{rep},i}$  using Eq. 3.3
end while

```

3.4 Simulation Methods

3.4.1 Target Representation

In this work, the targets do not emit a gradient field and agents are only able to detect a target's location when it lies within the target's radius. This makes the tracking problem more challenging by eliminating the possibility of using gradient-descent methods. This conservative approach represents one of the most challenging cases with a near-zero-range sensor tracking a target faster than the agent themselves. It is only through the deployment of an MRS in such challenging scenarios that it is able to fully make use of its swarm intelligence.

In this chapter, targets following a non-evasive movement policy are studied. Using this policy, a target generates and moves towards a random waypoint within the search area at a constant speed faster than that of its pursuing agents. When the target reaches this waypoint, it generates another random waypoint and moves towards it. This continues until the simulation is terminated.

3.4.2 Problem Statement

In this work, a swarm of tracking agents, $A = \{a_1, a_2, \dots, a_N\}$, and a set of targets, $O = \{o_1, o_2, \dots, o_M\}$, move within a bounded two-dimensional square search space of dimensions $L \times L$, where $L = 25$. Both agents and targets have an associated x and y position, $\mathbf{x}_i = (x_i, y_i)$, and maximum velocities of $\mathbf{v}_{a,\max}$ and $\mathbf{v}_{o,\max}$ respectively. The targets are able to move faster than the agents and as such, $\mathbf{v}_{a,\max} \leq \mathbf{v}_{o,\max}$.

In the case where a heterogeneous swarm is used, agents can be either slower agents or faster upgraded agents. These have associated maximum velocities of $\mathbf{v}_{s,\max}$ and $\mathbf{v}_{f,\max}$ respectively, where $\mathbf{v}_{s,\max} > \mathbf{v}_{f,\max} > \mathbf{v}_{o,\max}$.

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

$$\text{cov}(o_m, t) = \begin{cases} 1 & \exists i \in A \text{ s.t. } \|\mathbf{x}_i - \mathbf{x}_m\| \leq r, \\ 0 & \text{otherwise.} \end{cases} \quad (3.4)$$

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

$$\Xi = \frac{1}{TM} \sum_{t=1}^T \sum_{m=1}^M \text{cov}(o_m, t), \quad (3.5)$$

where T is the total time period of interest and M is the total number of targets within the search space. In the simulations performed, 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.

3.4.3 Swarm Performance Metrics

To quantify the overall tracking performance of the swarm, as well as its EED, several metrics were developed. The choice of a particular metric is an important one as it allows for the performance of the system to be compared for a given task. Currently, most of the metrics in the literature are very task specific and tend to focus on the action of one or a few particular agents, and not the swarm as a whole (Brambilla et al., 2013; Kolling et al., 2016). Furthermore, there are very few metrics in place used to quantify and perform analysis on a swarm's EED. As such, to compare the respective performances of various approaches a new suite of performance metrics were developed and proposed.

Tracking Performance

To measure the tracking performance of the swarm, the reward function presented in Eqn. (3.5) is used. This metric determines the proportion of time that a target has at least one agent within its radius. As such, the amount of time that all the targets are being tracked can be calculated. In the case of multiple targets being tracked, the

problem is similar to the Cooperative Multi-Robot Observation of Multiple Moving Targets problem proposed by Parker and Emmons (1997) and Parker (2002).

Exploration and Exploitation Dynamics

To quantify and permit the analysis of a swarm's exploration and exploitation dynamics, three separate metrics were developed. These are a swarm's cumulative velocity fluctuation magnitude, its heading-bearing correlation, and the system engagement ratio. It should be noted that in the study of tracking multiple targets, only the system engagement ratio metric is used as the cumulative velocity fluctuation magnitude and the heading-bearing correlation does not accurately quantify the swarm's EED in the presence of multiple targets.

Cumulative Velocity Fluctuation Magnitude In the seminal work done by Cavagna et al. (2010), it was found that the effectiveness of the transmission of information between birds in starling flocks is closely connected to the level of velocity fluctuations of individual agents and their correlations. The same conclusion was reached by Attanasi et al. (2014) when studying swarms of midges. Therefore, as a proxy for its collective activity and response to a dynamic stimulus, the level of velocity fluctuations in the proposed artificial swarm is quantified.

The velocity fluctuation of an agent i is given by the difference between its velocity and the mean velocity of the entire swarm:

$$\mathbf{u}_i[t] = \mathbf{v}_i[t] - \langle \mathbf{v}_j[t] \rangle_{j=1, \dots, N} = \mathbf{v}_i[t] - \frac{1}{N} \sum_{j=1}^N \mathbf{v}_j[t], \quad (3.6)$$

where $\mathbf{v}_i[t]$ is the velocity of an Agent i at a given time-step t . The overall response of the swarm to the dynamic environment can then be calculated by taking the time-averaged sum of all of the swarming agents' velocity fluctuations, divided by the number of agents in the swarm, and then normalising the value with respect to the agents' maximum speed:

$$\Gamma = \frac{1}{T_f} \sum_{t=1}^{T_f} \left\langle \frac{\mathbf{u}_j[t]}{v_{\max}} \right\rangle_{j=1, \dots, N} = \frac{1}{NT_f v_{\max}} \sum_{t=1}^{T_f} \sum_{j=1}^N \mathbf{u}_j[t], \quad (3.7)$$

where T_f is the total number of time-steps considered. In the case of heterogeneous swarms, the swarm's average maximum speed was used as the value for $v_{\max}$, defined as:

$$v_{\max} = \frac{N_f v_{f,\max} + N_s v_{s,\max}}{N}, \quad (3.8)$$

where $v_{f,\max}$ and $v_{s,\max}$ are the maximum speeds of the fast and slow agents respectively.

Heading-Bearing Correlation The heading-bearing correlation of a swarming system's agents introduced to quantify the level of exploratory and exploitative actions carried out by the MRS. This was done by calculating the correlation between the heading of any agent i and the target's bearing relative to this agent. This correlation can be calculated as follows

$$\phi_i[t] = \frac{\mathbf{v}_i[t] \cdot \mathbf{t}_i[t]}{\|\mathbf{v}_i[t]\|} = \cos \theta_i[t], \quad (3.9)$$

where $\mathbf{t}_i[t]$ is the bearing of the target in relation to the agent and $\theta_i[t]$ is the angle between the two unit vectors (see Fig. 3.1). With the definition of ϕ_i in Eqn. 3.9, ϕ_i will take on a value between -1 and $+1$. Agents associated with a ϕ_i value approaching -1 are carrying out more exploratory actions, while those with a ϕ_i value approaching $+1$ are carrying out exploitative actions by moving towards the target.

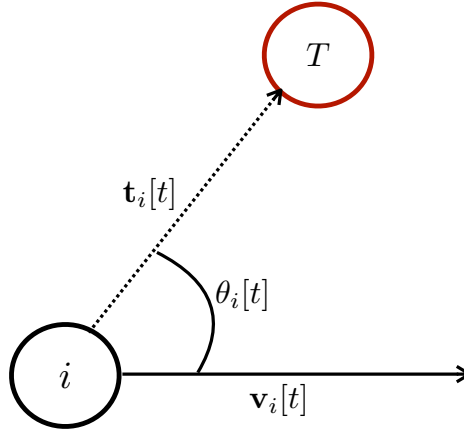


FIGURE 3.1: Bearing of the target ($T[t]$) relative to the agent ($\mathbf{t}_i[t]$) and the heading of the agent ($\mathbf{v}_i[t]$) at time t .

System Engagement Ratio The metrics that make use of a swarm's cumulative velocity fluctuation magnitude and heading-bearing correlation required that there only be one target within the search space. In addition, the EED of two different systems can only be compared when visually comparing their associated histograms; it does not facilitate the analysis of EED trends when system parameters are varied. As such, they were unsuitable for use when dealing with a swarm attempting to track multiple targets and a new metric that calculated the overall level of engagement of a system with targets was proposed, allowing for the system's EED to be quantified.

This metric makes use of an agent's tracking state, previously introduced in Section 3.3.3. An agent is considered to be engaged with a target if it has entered a tracking state and $S_i(\mathbf{x}_i[t], t) = 1$. Therefore, the overall engagement ratio of the swarm can be calculated

as follows:

$$\Theta = \frac{1}{NT} \sum_{t=1}^T \sum_{i=1}^N S_i[t], \quad (3.10)$$

where N is the total number of agents within the swarm. With this metric, a higher engagement ratio will indicate that agents are spending a higher proportion of time attempting to track a target. Conversely, at lower engagement ratios, agents spend a lower proportion of time attempting to track a target, which is characteristic of higher levels of exploration.

3.4.4 Simulation Setup

A two-dimensional square search-space (dimensions $L \times L$, where in this case $L = 25$) devoid of obstacles was considered with a disc-shape target having a fixed radius $r = L/25$. Up to 3 targets move randomly about the search-space at constant speed. If an agent falls within the radius of the target, given information about the target's position. This models a binary objective function, in which agents either have are fully informed of the target's position or have no information at all.

In the simulations performed, two MRS swarms were considered: (1) a homogeneous swarm consisting of only slower agents, and (2) a heterogeneous swarm consisting of both slow and upgraded fast moving agents. Slow agents were modelled with a maximum speed of $v_{a0,\max} = 0.1$ non-dimensional units per time-step and the fast agents are modeled with a maximum speed of $v_{a1,\max} = 0.26$ non-dimensional units per time-step. All simulation runs lasted for a total of 100,000 time-steps each to ensure statistically steady conditions. With such long simulation runs, low variability was calculated for the performance metrics ($< 1\%$) for similar runs using different random seeds. Figures with the uncertainty ranges can be found in Appendix C. For all simulations performed in this chapter, a total of 50 agents were used. These agents were randomly positioned within the search space during the initialisation of the simulation.

3.5 Simulation Results

3.5.1 Homogeneous Swarms

Cumulative Velocity Fluctuation Magnitude

The effects of the proposed strategy on an MRS can first be observed when looking at the EED of the system. In using the cumulative velocity fluctuation magnitude, the collective response of the swarm to dynamic changes in the search-space can be quantified. In the target tracking scenario posed in this work, these dynamic changes are represented by the fast-moving target. As such, the use of the cumulative velocity fluctuation magnitude can be considered as a proxy for a quantitative measure of the swarm intelligence activity. As seen in Fig. 3.2, there is a significant increase in the swarm's activity and response to the environment while using the adaptive repulsion scheme as compared to constant repulsion case. This is due to the new strategy

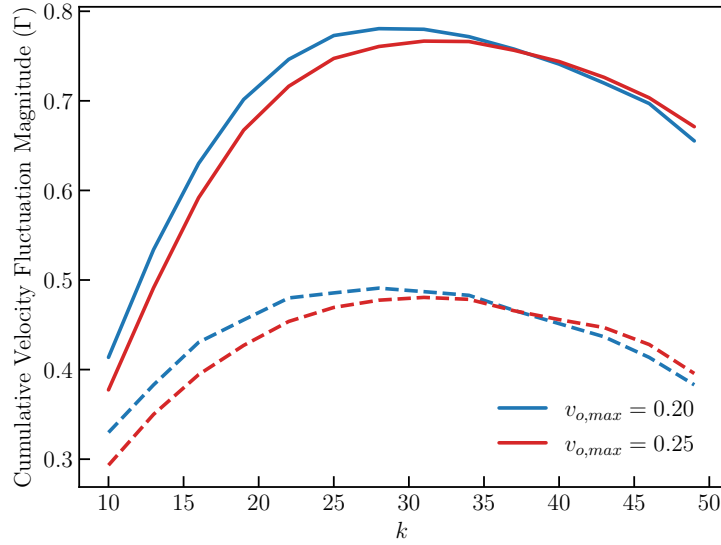


FIGURE 3.2: A system's cumulative velocity fluctuation magnitude, Γ , with varying levels of connectivity, k , and at two different target speeds, v , while using adaptive repulsion (solid line) and constant repulsion (dashed line).

promoting area exploration and preventing the agents from over-exploiting outdated target information.

The curves in Fig. 3.2 also illustrate a peak in system's response at intermediate levels of connectivity. The peak is more clearly marked for the collective response obtained with the adaptive repulsion scheme. It is interesting to note that there is a slight increase in the k value required to maximise a system's response when the target velocity increases. This can be explained as when the swarm encounters dynamic changes at different time-scales, it needs to adjust its communication channel to optimally respond. In this target tracking scenario, the swarm needs to adjust its level of inter-agent communications to optimally respond to targets of different speeds. Moreover, to generate the highest level of fluctuations in the swarm, as many agents as possible must be switching as often as possible from exploitative actions to exploratory ones. This can only happen at intermediate levels of connectivity.

Heading-Bearing Correlation

Next, the distribution of the heading-bearing correlation, $\phi_i[t] = \cos \theta_i[t]$, for all agents i in the swarm and over the entire duration of the simulation is considered. This allows for a histogram to be plot, allowing for the visualisation of the heading-bearing correlation of all agents within the swarm, and permits the visualisation of the swarm's EED. This is done by recording the heading-bearing correlations for all agents and iterations. The weight of each bin was then divided by the total number of iterations to give a time-averaged histogram for the entire swarm.

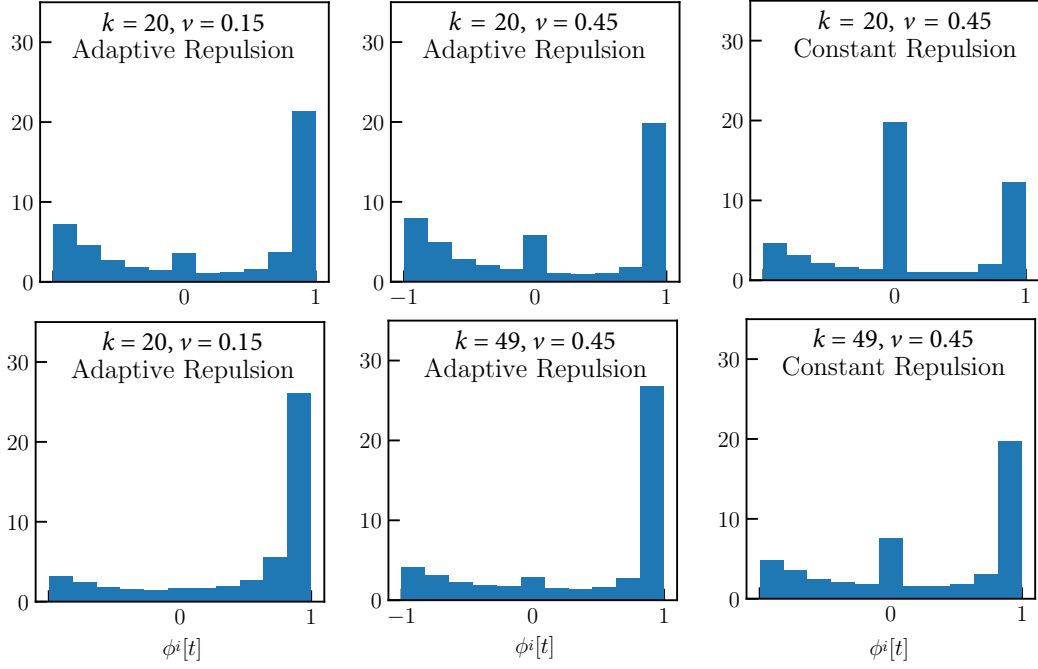


FIGURE 3.3: Distribution of the heading-bearing correlations, $\phi_i[t] = \cos \theta_i[t]$, (x -axis) for different levels of the swarm connectivity, k , target speeds, v . “Adaptive Repulsion” corresponds to a swarm utilising the newly proposed repulsion scheme, while “Constant Repulsion” considers the minimal static repulsion at $a_{R,\min}$.

Three observations can be made from Fig. 3.3. Firstly, it can be seen that as the level of connectivity, k , increases, the level of exploration of the environment decreases in favour of an increase in exploitative actions, regardless of the type of repulsion used. This shift toward a more exploitative collective behaviour when increasing k is robust and observed for all target speeds, v , considered. The change in EED balance towards exploitation is characterised by a higher proportion of agents having large and positive heading-bearing correlations, thereby indicating that more agents are moving toward the target rather than exploring the environment. While agents spending a larger proportion of time carrying out exploitation of the target may at first seem preferable, it is actually detrimental as the simulation unfolds. At high levels of exploitation, units tend to aggregate and cluster near the target’s position. Eventually, when the target ‘outruns’ its pursuing agents, it is able to easily move into portions of the search-space void of agents, thereby evading detection and reducing a system’s tracking performance.

Secondly, the amount of exploitation carried out by the swarm decreases while the amount of exploration increases with higher target speeds, v . This is characterised by the reduction in the number of agents with a positive heading-bearing correlation. The increased exploration is caused by the swarm losing track of the target more frequently, causing the swarming agents to disperse from the target’s last known location.

Finally, it can be seen that there are agents with a heading-bearing correlation value of 0. This is more prominent at higher velocities and when using constant repulsion.

This is caused by agents that are stationary or travelling at very low speeds. The reason for the near stagnation of the agents depends on the type of repulsion used. With the constant repulsion strategy, when the swarm loses track of the target, the agents hold their positions and remain in the vicinity of the target's last known location until target crosses the path of the agent or one of its neighbours. This is characteristic of a system over-exploiting outdated information. In contrast, while using the adaptive repulsion strategy, the agents disperse from the target's last known location. However, their movement is eventually restricted by the search-space boundaries and the blocking effect caused by a combination of a high repulsion strength and the presence of neighboring agents. The stagnant agents using the two repulsion strategies can be seen in Fig. 3.4.

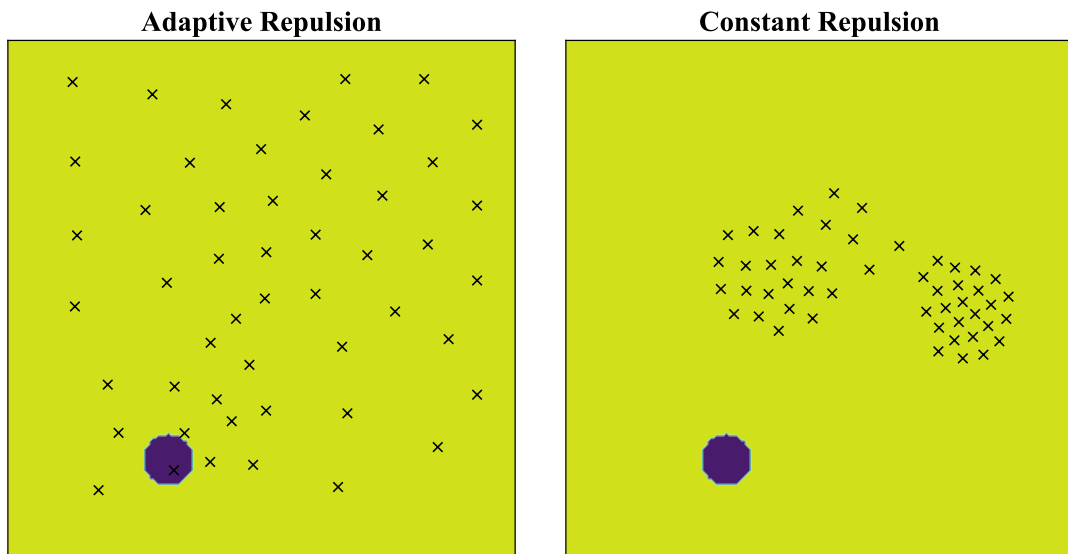


FIGURE 3.4: (Left) Screenshot of a simulated system using the adaptive repulsion scheme. Agents in the upper portion of the environment are being blocked by the search-space boundaries and neighboring agents. (Right) Screenshot of a simulated system using the constant repulsion scheme. All agents remaining in place after losing track of the target with the low exploratory behaviour in the presence of low constant repulsion.

Engagement Ratio

As mentioned previously, the usage of a system's heading-bearing correlation is incompatible with MRS tracking multiple targets. This is because a specific target cannot be assigned to an agent for the calculation of this metric. As such, the engagement ratio of a swarm is introduced as a means to quantify a swarm's EED, especially when tracking multiple targets.

As seen in Fig. 3.5, using the system's engagement ratio as a metric demonstrates that increasing the MRS' level of connectivity results in increasing levels of exploitation. The level of engagement, and by extension the total amount of exploitation carried out by the system, also decreases when the target speed is increased. These two observations

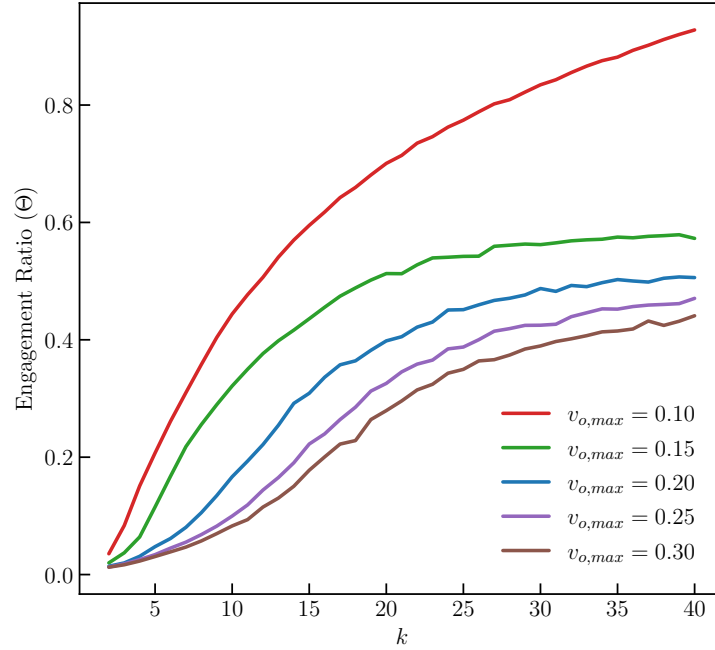


FIGURE 3.5: Swarm system engagement ratio, Θ , while tracking a single non-evasive target traveling at different speeds. Maximum agent velocity set at $v_{a,\max} = 0.1$.

are consistent with the conclusions obtained through analysis of the heading-bearing histograms seen in the previous section.

In addition, this plot reveals that when the target travels faster than its pursuing agents, there is an upper limit on the system's engagement ratio that decreases as the target's speed increases. This limit occurs when $k \rightarrow N - 1$ and when the target speed exceeds that of the maximum agent speed, i.e., $v_{o,\max} > 0.2$. This is because at higher levels of connectivity, the swarm loses track of the target easily due to a high portion of agents aggregating around and exploiting the target's location. Eventually, when the agents are outrun by the target, they have to expand and explore the environment again to relocate the target due to the agents' policy to explore the area if they are not trying to perform target tracking. This suggests that the amount of exploitation that can be performed by the swarm is being limited by the lack of information gathered by the swarm about the target's location, i.e., a lack of exploration, and further stresses the critical need to balance the amount of exploration and exploitation carried out by the swarm.

While tracking multiple targets, Fig. 3.6 shows that increasing the number of targets elicits a higher level of engagement, and hence exploitation, from the swarm. This is because the presence of more targets within the search space results in higher chances for an agent to encounter a target. This higher rate of target encounters inevitably lead to agents spending more time in the tracking mode to exploit target information, hence causing an overall increase in the engagement ratio.

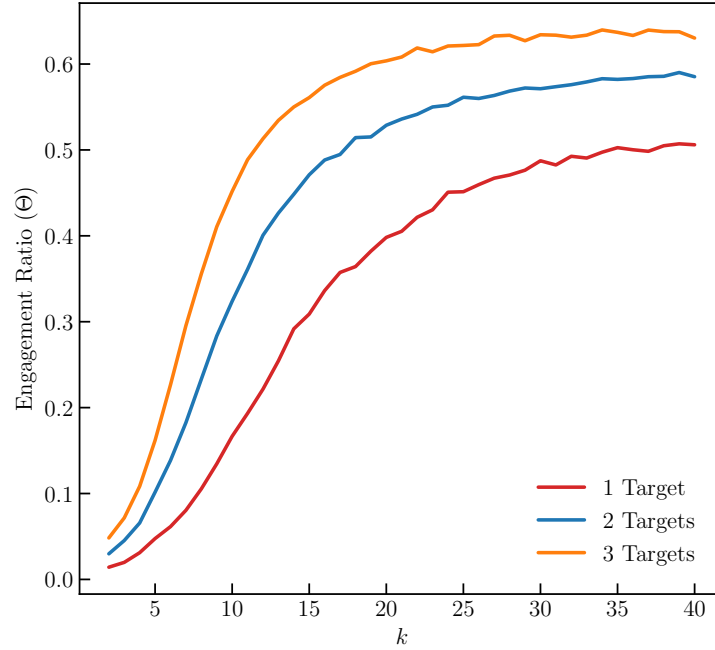


FIGURE 3.6: Swarm system engagement ratio, Θ , while tracking multiple non-evasive targets traveling at different speeds. Maximum agent velocity set at $v_{a,\max} = 0.1$. Maximum Target velocity set at $v_{o,\max} = 0.2$.

Tracking Performance

Single Target Tracking With the effects of the strategy of the MRS' EED demonstrated, its effects on the system's overall tracking performance can be explained. Firstly, as seen in Fig. 3.7, which plots the performance of a swarm tracking single non-evasive targets travelling at different speeds, increasing the target's speed had the effect of reducing a swarm's tracking performance across all values of k . This is true for both adaptive and constant repulsion schemes and is a direct consequence of the target's speed advantage over any individual agent, thereby allowing it to 'outrun' its pursuing agents. This is true for both adaptive and constant repulsion schemes.

Secondly, the figure also shows that the adaptive repulsion behaviour adopted yields a significant improvement over the more commonly used constant repulsion scheme, especially when operating at higher levels of connectivity. This improved tracking capability of the system stems from the increased exploratory behaviour promoted by the proposed adaptive repulsion scheme. While using this repulsion scheme, the swarm exhibits an improved collective search behaviour by expanding their formation, attempting to cover the entire search environments upon losing track of the target. In turn, this notably shortens the time required to re-engage the target. As already highlighted, the reduced exploratory dynamics seen when using the constant repulsion scheme leads to the swarm aggregating and then stagnating after being outrun by the target, thereby resulting in poor tracking performances.

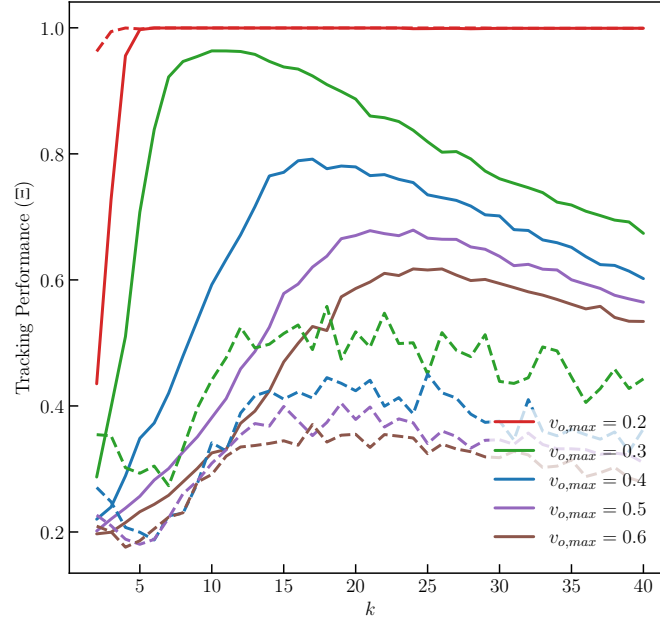


FIGURE 3.7: Tracking performance of a swarm consisting of 50 agents pursuing a single non-evasive target travelling at different velocities. Swarming agents travelled at a maximum speed of $v_{a,\max} = 0.1$. The solid line represents a swarm utilising the adaptive repulsion scheme with a variable a_R , while the dotted line represents a swarm utilising a constant repulsion scheme with a constant a_R .

Finally, Fig. 3.7 reveals the existence of an optimal degree, k^* , for the interaction network at which the tracking performance peaks. This optimum only occurs when the target is able to move faster than the agents. Starting from $k = 2$, an increase in the level of connectivity among swarming agents improves up to k^* . Beyond this value, more connectivity ends up being detrimental to the collective tracking performance. This is because at low target velocities, the swarm need not regulate its level of exploration and exploitation to effectively track the target; each agent is able to continuously exploit the target's positional information without the aid of its neighbours. In contrast, when the target travels faster than the agents, the agents need to cooperate to effectively perform the tracking task. As such, members of the swarm need to continuously switch between exploratory and exploitative actions to reacquire the target's location after being outrun to eventually re-engage with the target.

Interestingly, these peaks in tracking performance are not as apparent in the swarm using the more commonly used constant repulsion scheme. However, it can still be observed that higher levels of connectivity using this scheme are detrimental to the system's collective tracking performance. From these results, it can be concluded that too high a level of connectivity induces over-exploitation in a system's agents that is counterproductive to the overall performance of the swarm. In contrast, at low levels of connectivity, the system carries out excessive levels of exploration and under-exploits target information, also leading to sub-optimal performances.

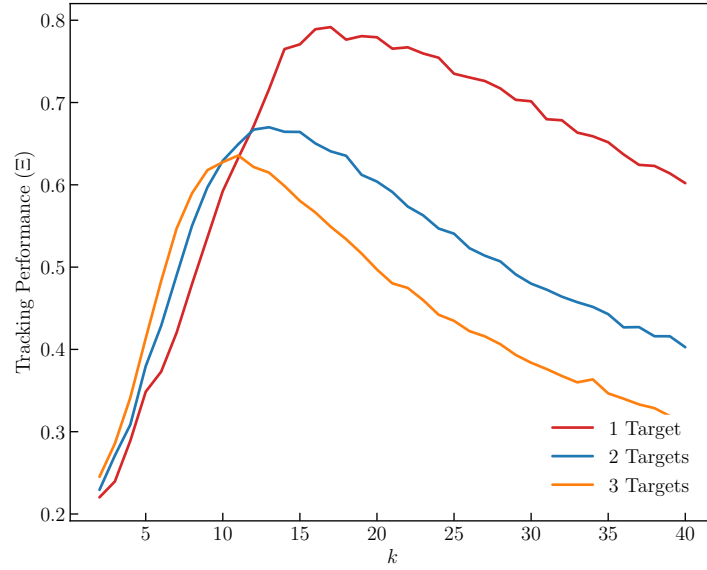


FIGURE 3.8: Tracking performance of a swarm pursuing multiple non-evasive targets with a speed of $v_{o,\max} = 0.2$. Pursuing agents travelled at a maximum speed of $v_{a,\max} = 0.1$.

Multi-Target Tracking As can be expected, increasing the number of targets to be tracked reduces the overall tracking performance of the system, as seen in Fig. 3.8, because the MRS is expected to simultaneously pursue all targets in the environment. In addition, it can be seen in the figure that a system's k^* reduces as the number of targets to be tracked increases due to system's increased need for exploration. The need for more exploration is due to two reasons: (1) the need to search for the additional targets, and (2) greater number of targets illicit higher levels of exploitation from the swarm. As previously discussed, the higher levels of exploitation in the presence of more targets within the search space is due to the higher probability of an agent encountering a target while exploring the search space. This higher probability of target encounters leads to agents spending more time tracking targets and therefore higher levels of exploitation. Therefore, the swarm requires a greater amount of exploration carried out to balance this increase in exploitation and find the locations of the other target, thereby maximising its tracking performance in the presence of multiple targets. In the presented strategy, this greater need for exploration is satisfied at lower levels of connectivity, hence leading to a reduced k^* .

In addition to the higher level of exploration attained at lower values of k , utilising a network with lower levels of connectivity also results in the formation of multiple clusters of agents, which is also a characteristic of high levels of exploration. Since each cluster tends to track only one target, the presence of multiple agent clusters facilitates the tracking of more than one target simultaneously. This property is demonstrated in Fig. 3.9.

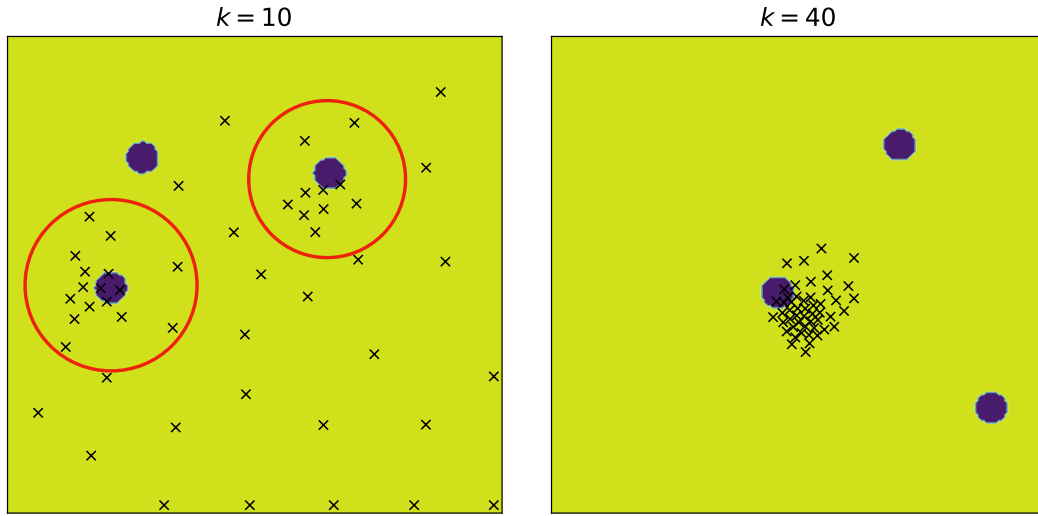


FIGURE 3.9: Distribution of agents (black crosses) while tracking multiple targets (blue circles). Agents using a lower level of connectivity more readily form multiple clusters, circled in red, to track targets (left). Agents with higher levels of connectivity tend to form one large cluster that can only track a single target (right).

3.5.2 Strategy Comparisons

Having ascertained its tracking performance, the newly developed strategy is compared to three other benchmark strategies. These benchmark strategies are: (1) $k = 0$ —corresponding to the case where there is no inter-agent communication and cooperation, (2) $k = 49$ —corresponding to the case where a global communications network is used, and (3) the standard PSO algorithm proposed by Kennedy and Eberhart, 1995. These results were obtained by simulating the system tracking a single fast-moving non-evasive target. The simulations were repeated 20 times for each of the four different strategies with different random seeds.

Figure 3.10 shows that by using the newly proposed strategy, a multi-agent system is indeed able to demonstrate a higher tracking performance when its level of connectivity has been optimised. This further emphasises the importance of attaining the correct balance between exploration and exploitation being performed by the system, which can be tuned by varying the system's level of connectivity. In addition, the figure also illustrates that the newly proposed strategy is able to outperform the base PSO strategy regardless of the system's degree of connectivity. This serves to highlight the main weakness of swarms using the original PSO algorithm to track fast-moving targets as stated by Jordehi, 2014: (1) the swarm continuously uses outdated information, causing agents to aggregate in a location where the target is no longer present, and (2) the swarm constantly tries to exploit the target's positional information without exploring the environment.

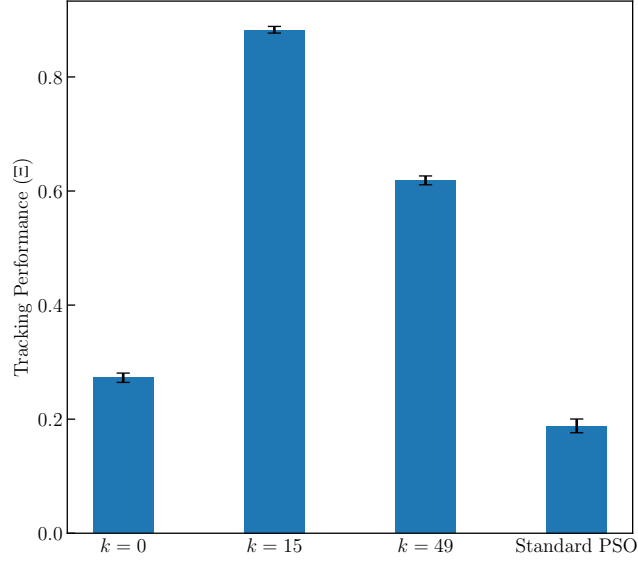


FIGURE 3.10: Tracking performance of swarms using the proposed search and track strategy with (1) a $k = 0$ network, (2) a $k = 15$ network, and (3) a $k = 49$ network, as well as a swarm using the standard PSO algorithm. The target moved at a maximum speed of $v_{o,\max} = 0.2$ while the pursuing agents travelled at a maximum speed of $v_{a,\max} = 0.1$.

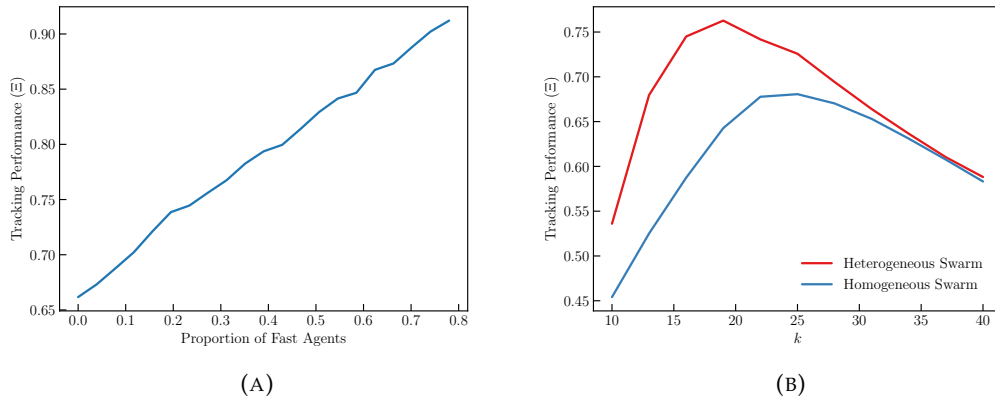


FIGURE 3.11: Tracking performance of a heterogeneous swarm pursuing a non-evasive target moving at a speed of $v_{o,\max} = 0.3$. Slow agents move at a speed of $v_{s,\max} = 0.1$ and fast agents move at a speed of $v_{f,\max} = 0.26$. (A): A heterogeneous system with a level of connectivity, k , is maintained at $k = 20$ while the proportion of fast agents in the swarm is varied. (B): A heterogeneous system where the agent composition of 15 fast agents and 35 slow agents compared to a homogeneous system comprised of 50 slow agents. The performance of the two systems are compared while k is varied.

3.5.3 Heterogeneous Swarms

Tracking Performance

In this section, the use of the proposed strategy in heterogeneous systems is studied. As previously demonstrated by Altshuler (2009), the use of heterogeneous systems with upgraded agents does not guarantee the improvement of system performance. However in the target tracking scenario considered in this work, the results visualised in Fig. 3.11a show that the overall tracking performance of the swarm increases with the proportion of fast buoys included within the swarm and is consistent across all levels of connectivity. This result can be attributed to the faster buoys being better able to keep up with and respond to the presence of the fast moving target. Similar to the homogeneous swarm, there exists an optimum degree of connectivity, $k^* \sim 18$, at which the tracking performance is at a maximum for the heterogeneous swarm, as evidenced by Fig. 3.11b. Beyond this point, the benefit of adding upgraded units decreases at higher levels of connectivity. Again, this is due to the system's tendency to over-exploit target information, leaving it vulnerable to being outrun by the target. Interestingly, this optimum occurs at a lower level of connectivity compared to the one for the slower homogeneous swarm. This occurs because of the higher mobility of the faster agents, allowing it to better track and pursue with the target. In addition, this higher level of mobility gives the swarming agents a greater amount of responsiveness, which will be explained in the next section.

Cumulative Velocity Fluctuation Magnitude

Similar to the heightened levels of tracking performance, Fig. 3.12 shows that a swarm's responsiveness gradually increases with an growing proportions of fast-moving agents. The result stems from the upgraded agents' ability to travel faster and keep up with the target. This better tracking ability in turn allows the fast agents to broadcast the target's position for longer, thereby allowing the system to maintain a high level of responsiveness.

In addition, Fig. 3.13 illustrates that there is a peak in the systems' levels of response at intermediate levels of connectivity in both homogeneous and heterogeneous swarms. This peak occurs at a lower degree in heterogeneous swarms ($k \sim 25$) than in the homogeneous one ($k \sim 30$). This is because upgrading a portion of the swarming MRS with faster agents essentially increases the average speed of the swarm. Therefore, even though the environment evolves at the same speed, this evolution occurs at a different time-scale relative to the swarm. By adding faster agents, the swarm needs to readjust its communications channel to adequately and optimally respond to its environment.

It should also be noted that in both homogeneous and heterogeneous swarms, the optimum level of connectivity associated with maximum response is higher than the level of connectivity required for maximum tracking performance. When the MRS uses a network that maximises its level of collective response, a large proportion of the swarm is attracted to the target once it is found. Consequently, there are little to no agents remaining to perform area exploration. This leads to the target eventually outrunning the system. Conversely, when the swarm's communications network is connected to

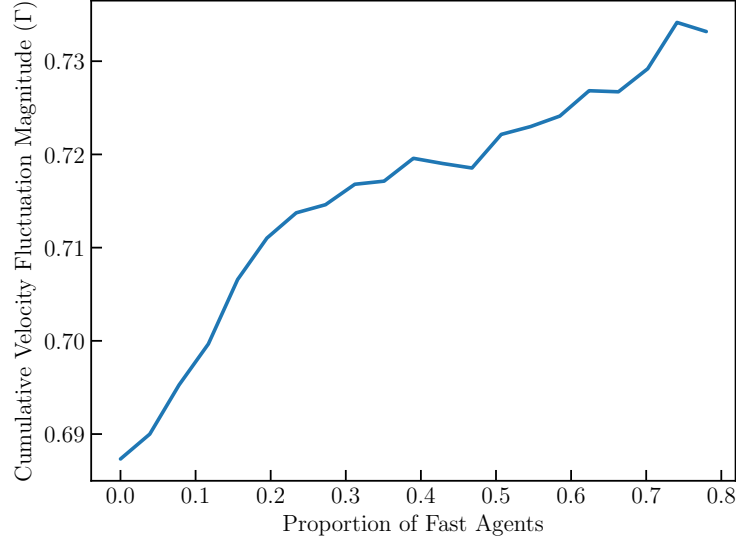


FIGURE 3.12: The cumulative velocity fluctuation magnitude, Γ , of a swarm with varying proportions of fast agents using a network degree of $k = 20$. 50 agents were simulated tracking a single non-evasive target travelling at a speed of $v_{o,\max} = 0.3$. Slow agents moved at a speed of $v_{s,\max} = 0.1$ and fast agents moved at a speed of $v_{f,\max} = 0.26$

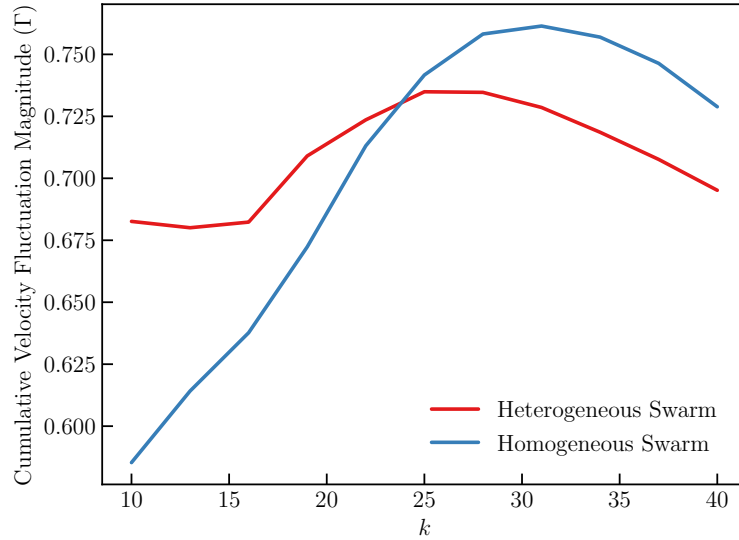


FIGURE 3.13: The cumulative velocity fluctuation magnitude, Γ , of a swarm of 50 agents comprised of 15 fast agents with varying levels of connectivity, k . The system tracked a single non-evasive target travelling at a speed of $v_{o,\max} = 0.3$. Slow agents moved at a speed of $v_{s,\max} = 0.1$ and fast agents moved at a speed of $v_{f,\max} = 0.26$

provide maximum tracking performance, there is a proportion of agents that have not received information on the target's position. These agents do not actively move towards a target and instead perform exploratory actions. Hence, when the initial group of pursuing agents are outrun, this group of exploratory agents are able to continue tracking the target.

Heading-Bearing Correlation

As with the homogeneous swarm, the distribution of the heading-bearing correlation, $\phi_i[t]$, for all agents i in the swarm is visualised for the entire duration of one complete simulation. This distribution permits the creation of a histogram, thereby allowing the analysis of the system's EED and can be seen in Fig. 3.14. It should again be noted that analysis on a system's EED using the heading-bearing correlation can only be made when tracking a single target.

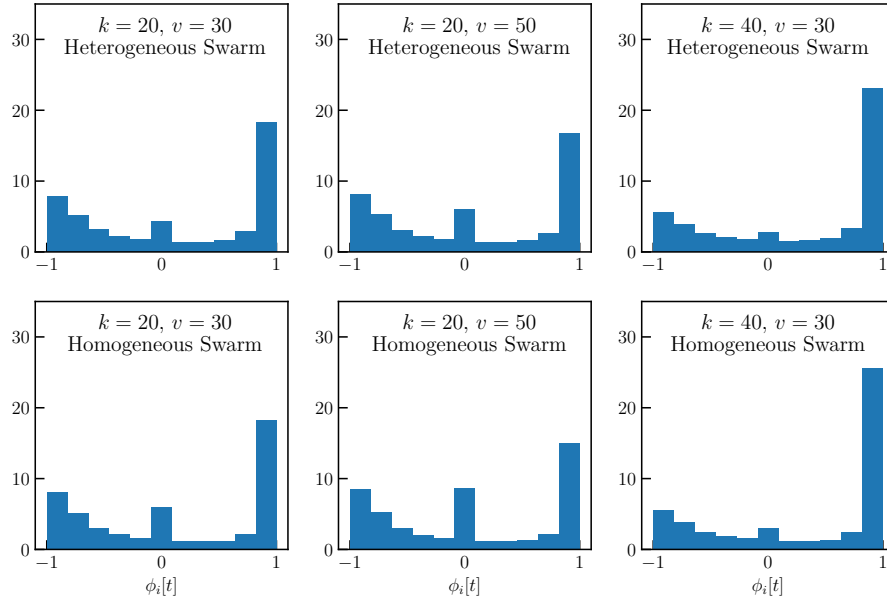


FIGURE 3.14: Distribution of heading-bearing correlations, $\phi_i[t]$, for different levels of the swarm connectivity, k , and target speeds, v . The upper row shows the heading-bearing correlations for a heterogeneous swarm comprised of 15 fast agents and 35 slow agents, while the lower row shows the heading-bearing correlations for a homogeneous swarm of 50 slow agents. From the histograms, it can be seen that, as with the homogeneous swarms, when the level of connectivity increases, so too does the amount of exploitative behaviour carried out by the swarm. This increase in exploitation is characterised by a higher proportion of agents having a value large positive heading-bearing correlation value, indicative of a higher proportion of agents moving towards the target. This is due to the target's positional information being more readily disseminated throughout the system due to the introduction of fast-moving agents, thereby causing the agents to remain in the tracking state and move towards the targets for longer periods of time.

It can also be observed that in both homogeneous and heterogeneous swarms, the average proportion of agents with a heading-bearing correlation of 0 increases at higher target speeds, implying a greater number of stationary agents. Similar to the homogeneous case discussed previously, this high number of stationary agents occurs due to high levels of inter-agent repulsion that causes the agents to remain in position. When a target is able to move faster, it is able to evade the agents more easily. The absence of target information cause the agents to spread out across the search area and be ‘locked’ in a static equilibrium position when their repulsion strength reaches the maximum limit. The agents remain in this position until they receive new information of the target’s position (see Fig. 3.15).

However unlike the homogeneous case, when fast agents are introduced to the system, there is a small decrease the proportion of agents with a heading-bearing correlation of 0. This can be attributed to higher collective response of the heterogeneous swarm at lower levels of connectivity compared to the homogeneous swarm, which was discussed in the previous section. This higher level of response means that agents are react more readily to the presence of a target, thereby encouraging agents to move away from their equilibrium positions and preventing the swarm from stagnating.

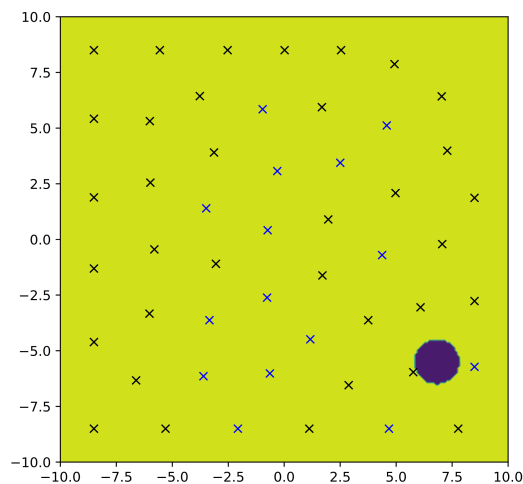


FIGURE 3.15: A heterogeneous swarm comprised of 35 slow agents (black crosses) and 15 fast agents (blue crosses) tracking a disc-shaped target (dark blue). Agents to the left of the target are locked in position due to them attaining their maximum repulsion strength and lack of target information.

3.6 Robotic Experiments

3.6.1 Swarm Robotic System

To validate the results obtained from the simulation, a decentralised swarm comprised of 8 Orion differential-drive land robots were used. Each robot operates through the use of two controlling software layers—a high-level layer and a low-level one. The

low-level layer is tasked with way-point navigation and estimating the robot's position. This is done through the use of two DC motors attached with encoders, as well as an Inertial Measurement Unit (IMU). The IMU allows each robot to estimate its position using dead reckoning. The robot's estimated position is then sent by the low-level layer to the high-level layer that grants the robot its autonomy as part of the swarming system. The high-level layer consists of a low-power and low-bandwidth wireless XBee-PRO module as well as a Raspberry Pi zero that is responsible for implementing and processing the required swarming behaviours and managing the information received from the neighbouring robots. This includes filtering and processing the data that is transmitted only from neighbouring units according to the selected value k of the inter-connecting topological network, which is selected before the running of each experiment. From the processed data, the high-level layer is able to generate a navigational way-point used by the low-level layer (Kit et al., 2019). Communications between the robots are facilitated by the XBee module that broadcasts and receives positional information and objective function values. The reader is directed to Kit et al. (2019) and Sundram et al. (2019) for further information concerning the technical specifications of the Orion robots.

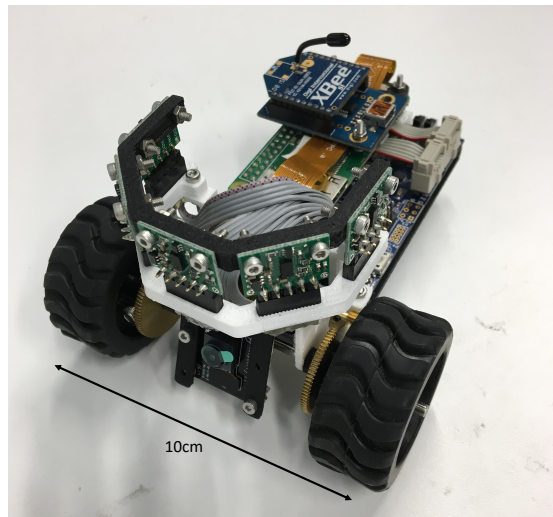


FIGURE 3.16: Orion robotic platform used in the swarm robotic experiments (Kit et al., 2019).

A fictitious sensing unit is programmed into the high-level layer to emulate the conservative scenario with a zero-range sensor capability used in the simulations. Sensory information—i.e. whether the agent is able to detect the target—is provided at every instant by the Raspberry Pi zero, based on the estimated position of the robot in relation to the prescribed trajectory of the moving target. Similar to the simulations performed, this virtual target moves randomly in the search-space at a predetermined speed, v .

3.6.2 Experimental Conditions

At the start of each experimental test, 8 robots are arranged in two lines in the middle of an open search-space (see Fig. 3.17, left column). All the robots broadcast their estimated position within the search space, as well as their tracking state. A target shaped as a disc of radius 1 m is simulated to be in constant random motion around the search area. The robots and the target travel at the same speed of $10 \text{ cm} \cdot \text{s}^{-1}$. It is important to note, however, that the fictitious target has an infinite maneuverability compared to the finite one of our robots. Hence, even if the individual robots have the same speed as the target, they are effectively much slower to respond changes in their heading owing to their low maneuverability. In practice, the dynamic target is implemented by generating a list of target positions and having the robots calculate the target's position based on the elapsed time. The robots then compare their respective position against the target's position and to verify if they are able to detect the target.

The experiments were run for two different connectivity levels ($k = 3$ and $k = 6$) and using both the constant and adaptive repulsion schemes. Each test is stopped after one minute and 5 repetitions were performed for each case. It is assumed for these tests that all robots are able to perfectly estimate their location while moving around the operating space.

It is important stressing that, aside from the logging of data for the path reconstruction of the robots, the robotic test platform operates in the complete absence of any supporting infrastructure (Kit et al., 2019). In all the experiments, the localisation, cooperative control, and computing tasks are performed by the individual robotic units in a fully decentralised manner.

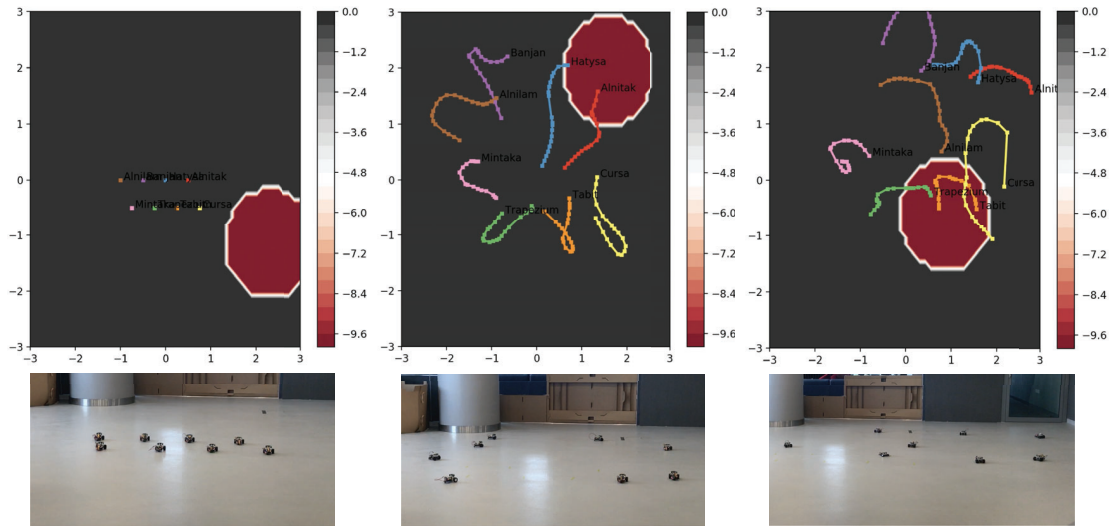


FIGURE 3.17: (Top) Positions of the target, represented by the red circle, the robots, and path reconstruction of the robots as observed through a central monitoring system, represented by the coloured dots and their associated trails. (Bottom) Photos from the physical experiment using a connectivity of $k = 6$ and adaptive repulsion. Each column corresponds to a different instant during the collective search and tracking run.

3.6.3 Experimental Results

Qualitative Observations

As seen through the reconstruction of paths taken by the Orion robots (Fig. 3.17), the agents are able to disperse from their original positions within the starting lines thanks to the adaptive repulsion behaviour. When a robot detects the target, other agents that have included the detecting robot within their neighbourhood are able to respond, turn, and start moving towards the target, as illustrated in Fig. 3.17 (centre column). Should the detecting robot be outrun, another robot is able to continue the tracking task. The robot that initially detected the target is able to change its heading and move towards the target's location, as shown in Fig. 3.17 (right column).

Tracking Performance

TABLE 3.1: Tracking performance of the swarm robotic system.

Repulsion Type	$k = 3$	$k = 6$
Adaptive Repulsion	0.26 ± 0.20	0.17 ± 0.05
Constant Repulsion	0.31 ± 0.34	0.24 ± 0.13

Table 3.1 shows the percentage of time when there is at least one robot within the target's radius. This is presented together with the 99% confidence intervals, permitting the quantification of the swarm's tracking performance. From these results, it can be seen that there is no significant difference between the two repulsion schemes. While there seems to be a decrease in the tracking performance when going from $k = 3$ to $k = 6$, given the relatively high variability in the results for $k = 3$, these results were determined to not be of statistical significance.

Although visual observations of the collective dynamics showed noticeable differences for different k values and repulsion schemes, the quantitative results for the tracking performance are not congruent with the simulations. This can be attributed to three reasons: (1) the small number of robots used within the swarm and relative density of the robot swarm, (2) the short time duration of each test, (3) the relative high speed of the target given that the finite maneuverability of the robots as compared with the particles in the simulations. Of these reasons, the low swarm density is likely to be the main cause of this discrepancy. As demonstrated by Cziráok, Stanley, and Vicsek (1997), statistical characterization of a multi-agent system's order becomes more difficult at low swarm densities. This makes it harder to predict the behaviour of a system with low density within simulations.

System Response

From the same experimental results used to quantify the tracking performance, the time-averaged cumulative velocity fluctuations of the swarm, Γ , was calculated, and

hence the system's level of response to the dynamic target. Table 3.2 presents the calculated response levels, along with the 99% confidence intervals. These results unambiguously show that the adaptive repulsion scheme significantly increases the response of the swarm to the target ($p < 0.01$) for both connectivity levels considered. In addition, a noticeable drop was measured in collective response with both repulsion schemes when increasing the network connectivity from $k = 3$ to $k = 6$. This latter observation is fully consistent with the simulation results previously presented in Section 3.5.1.

TABLE 3.2: Response of the swarm robotic system to a dynamic target.

Repulsion Type	$k = 3$	$k = 6$
Adaptive Repulsion	1.36 ± 0.07	1.23 ± 0.03
Constant Repulsion	1.23 ± 0.03	1.05 ± 0.08

It should be noted that the time-averaged cumulative velocity fluctuation magnitudes obtained experimentally are higher than those obtained in the simulations. This was expected as the robots were unable to instantaneously change their speed and heading unlike the omnidirectional agents modelled by the simulations.

3.7 Discussion

In this chapter, a strategy based on a memory-less PSO algorithm was presented for use in decentralised MRS to track fast-moving non-evasive targets. This strategy relies on two key features: an adaptive inter-agent repulsion behaviour and a dynamic k -nearest neighbours communications network. These features are used to tune the amount of exploration and exploitation that is carried out by the system. The strategy was tested in both homogeneous and heterogeneous MRS. To characterise the performance and behaviour of the swarm, four metrics were established: (1) Cumulative velocity fluctuation magnitude, used to quantify the system's response, (2) heading-bearing correlation, used to quantify a swarms exploration and exploitation dynamics, (3) engagement ratio, also used to quantify a swarms exploration and exploitation dynamics, and (4) tracking performance, used to quantify the overall system performance.

With the proposed metrics and strategy, it was found that increasing the value k in the k -nearest neighbours communications network resulted in the swarm performing increasing amounts of exploitation. In contrast, reducing k resulted in increased levels of exploration. By conducting simulations using different values of k , it was revealed that there exists of an optimal value of $k = k^*$ that maximised the tracking performance of the system, highlighting the need to balance the amount of exploration and exploitation being performed by a swarm. The results obtained from the simulations were further validated through physical robotic tests carried out with a swarm of 8 decentralised land robots. Both sets of results challenge the commonly held belief that systems with higher connectivity perform better and are congruent with other studies that have also showcased the need to moderate the level of inter-agent communication within a system (Rausch et al., 2019; Horsevad, 2021; Talamali et al., 2021).

The newly developed strategy was also implemented within a simulated heterogeneous swarm. The results obtained show that the tracking performance of the system increases together with the proportion of fast agents within the system. This is mostly due to the faster agents being better able to keep up and respond to the presence of a target. While no changes are necessary to elicit these improvements, the performance of a heterogeneous swarm can further be optimised by reducing the system's level of connectivity, resulting in a lower k^* compared to the homogeneous system.

Besides the strategy, the work presented in this chapter shows the importance of balancing a swarm's EED according to the system's environment. While the system may be able to track a target to a limited extent at any degree of connectivity, the tracking performance is maximised only when the system is set to operate using the optimum level of connectivity, k^* . As previously mentioned, the value of this k^* varies according to the system's operating environment, such as the target's speed and the number of targets present. This shows that the optimal EED balance changes according to the task's demands, thereby suggesting that any developed swarm strategy must be tailored to the specific operating conditions to ensure optimum performance; one cannot expect a strategy developed specifically for a set of conditions, A, will work equally well when deployed in a different set of conditions, B.

Chapter 4

Evasive Targets

4.1 Introduction

Having developed a strategy that can be used by a multi-robot system (MRS) tracking non-evasive targets, work can now be done to adapt it for use in swarming MRS tracking fast-moving evasive targets. Similar to the problem with tracking fast-moving non-evasive targets, the task of tracking fast-moving evasive targets has not been well studied. This is especially the case when multiple targets are to be tracked long-standing assumption that the targets will always be able to evade their pursuers given their superior mobility and maneuverability. This resulted in the problem being determined to be impossible and therefore trivial (Parker and Emmons, 1997; Parker, 2002; Khan, Rinner, and Cavallaro, 2018).

In the current literature dealing with fast-moving evasive targets, works tend to focus on the capture of one (Shishika and Paley, 2019; Fang, Cheng, and Xie, 2020) or multiple targets (Janosov et al., 2017; Zhang et al., 2019a). Once an agent is within a specified radius surrounding a target, the target is considered to have been caught and is either immobilised or removed from the environment. As such, these works ignore cases where the target continues to move, even after it has been localised.

In this chapter, the search and track strategy proposed in Chapter 3 is adapted for use for the search and tracking of fast-moving evasive targets. In addition to the k -nearest neighbour communications network and the adaptive repulsion behaviour, this strategy is augmented by the use of an agent-based memory. The usage of agent-based memory is demonstrated to be critical in enabling a robotic swarm to track an evasive target by giving the MRS a more persistent point of attraction. The introduction

The content in this chapter has been published in the following papers:

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*. DOI: 10.1162/isal_a_00376

Kwa, H.L., Kit, J.L., and Bouffanais, R. (2021). "Tailoring Exploration and Exploitation in Multi-Agent Systems with Short-Term Memory and Limited Social Interaction" In: *2021 International Workshop on Agent-Based Modelling of Human Behaviour (ABMHuB)*

Kwa, H.L., Babineau, V., Philippot, J., and Bouffanais, R. [In Press]. "Adapting the Exploration-Exploitation Balance in Heterogeneous Swarms: Tracking Evasive Targets". In: *Artificial Life*. arXiv: 2207.13523

of memory also gives system designers another control parameter to adjust an MRS' exploration and exploitation dynamics (EED), which permits agents to prioritise either exploratory or exploitative actions. This new algorithm is studied through its implementation in simulated homogeneous and heterogeneous swarms.

Given that both the length of agent-based memory and level of connectivity used in a swarm affects its level of exploitation carried out, further analysis is carried out on the interplay between these two factors that give rise to the optimum amount of system engagement necessary to maximise the system's tracking performance. This strategy to carry out the search and tracking of evasive targets is validated through tests performed on a swarm comprised of six Orion robots. Finally, the possibility of deploying differentiated tracking strategies in heterogeneous swarms are explored.

4.2 Related Work

4.2.1 Chase and Capture Patterns

When capturing evasive targets, Kamimura and Ohira (2019) identified three patterns observed in groups of agents pursuing evasive targets (see Fig. 4.1). In their simulations, linear target chase (Fig. 4.1 A) was mostly observed in systems with small agent numbers. However, they also observed that to capture a target, all systems regardless of size tended to display target encirclement (Fig. 4.1B) or have an agent drive targets towards other pursuing agents (Fig. 4.1C).

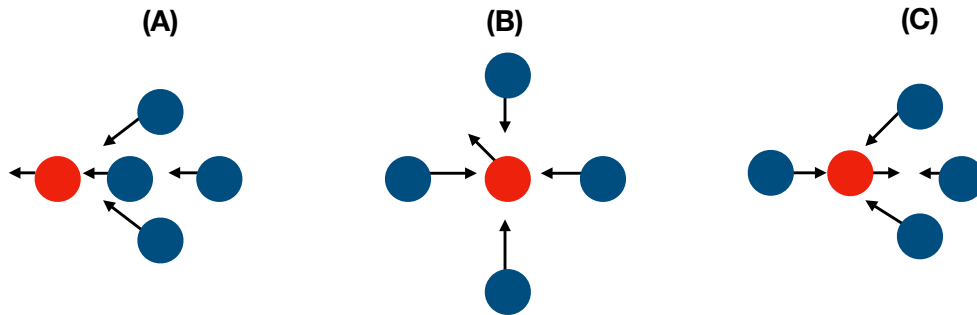


FIGURE 4.1: Common chase strategies used by agents (blue) while pursuing an evasive target (red). (A) Linear target chase, (B) Encirclement of target, and (C) Driving of target into other agents. (Figure adapted from Kamimura and Ohira (2019).)

These observations were also made by Janosov et al. (2017) in their simulations, where linear target chase was determined to be ineffective in capturing a faster target. Qi et al. (2020) also demonstrated that cooperative behaviours leading to the entrapment of a target lead to faster capture times. To encourage the target trapping behaviours seen in Figs. 4.1B-C, Zhang et al. (2019b) developed a *short-range repulsion, long-range attraction and medium-range repulsion* (SRLAMR) agent interaction strategy. This type of interaction caused agents to dynamically form unstable clusters of agents that eventually disperse. The constant formation and dispersion of clusters results in a higher chance of targets being entrapped by aggregating agents. Such strategies were also noted by

Ritz et al. (2021) when evolving predator strategies for speed handicapped pursuing agents while using reinforcement learning methods.

4.2.2 Agent-Based Memory

In most scenarios studied, the pursuing agents usually has a field of vision, allowing them to pursue targets outside of their capture or tracking radius (Janosov et al., 2017; Zhang et al., 2019b; Ritz et al., 2021). However, in this research work, the situation where pursuers have a zero-range sensor is studied — i.e. a pursuing agent must be in the same location as the target to detect its presence. In such situations involving zero-range sensors and fast-moving evasive targets, the use of memory is critical as it provides the chasing agents a longer lasting point of attraction, thereby allowing agents to exploit information regarding a target's location.

It has been established that memory may cause a multi-agent system (MAS) to exploit outdated information (Jordehi, 2014; Coquet et al., 2019). Despite these disadvantages, memory usage has been shown to encourage the aggregation of agents around high quality target patches in various scenarios. In various foraging tasks, it has been demonstrated that memory was required for agents to aggregate in the most profitable locations (Falcón-Cortés, Boyer, and Ramos-Fernández, 2019; Nauta, Simoens, and Khaluf, 2020). Furthermore, it was observed by Nauta, Khaluf, and Simoens (2020) that memory allowed their MAS to exploit the area surrounding the most salient locations, allowing the system to maximise the amount of reward collected.

When making use of memory, the length of memory and the amount an agent relies on memory needs to be calibrated according to the presented scenario. Saffre et al. (2019) observed that systems employing agents with long term memory tended to consume the same limited resource at the same time. This led to over-exploitation of the resource, leading to system congestion, and reduction of the system's performance. Nauta, Khaluf, and Simoens (2020) demonstrated that high target diversity favoured systems less dependent on memory, allowing agents to better explore the area of operation, while a scenario involving highly clustered targets favoured the use of agents that relied heavily on memory, allowing for revisiting of target clusters.

Related to the use of memory is the use of forced agent position resets. This was a strategy first pioneered in the field of computational optimisation (Carlisle and Dozier, 2000; Hu and Eberhart, 2002; Zhan et al., 2009) and made its way into swarm robotics (Jatmiko, Sekiyama, and Fukuda, 2007; Yan et al., 2018). Using these types of strategies, a central controller, which is endowed with memory, will command the agents to reset their positions to a random location within the search space after a point of interest has been exploited for a long period of time or when a large change is detected in the environment. The use of this type of strategy prevents the over-exploitation of any resource or point in the environment and allows the system to 'jump out' of local optima and encourages exploration when exploitation is no longer deemed useful.

4.3 Memory-Augmented Search and Track Strategy

The new memory-augmented strategy presented here uses the one presented in the previous chapter and builds on it by utilising memory-enabled agents. Compared to memory-less agents, memory-enabled agents are able to generate a more persistent point of attraction. This gives them more time to exploit a target's positional information and move towards the target's location. The use of memory is critical in the tracking of a fast-moving evasive target as it allows the system to converge on a target even though agents are unable to detect its presence. However, in contrast to other strategies based on the PSO algorithm that uses infinite memory, the proposed strategy is able to limit the use of outdated information by using short memory lengths.

Similar to the original strategy, the memory-augmented strategy consists of two behavioural patterns, i.e. one to promote agent aggregation and drive exploitation of information and another to promote inter-agent repulsion, encouraging exploration of the environment. These two behaviours generated two velocity vectors at each time-step, which were summed to give an individual agent's final velocity vector:

$$\mathbf{v}_i[t] = \mathbf{v}_{i,\text{att}}[t] + \mathbf{v}_{i,\text{rep}}[t], \quad (4.1)$$

where $\mathbf{v}_{i,\text{att}}[t]$ is the velocity vector generated by the attractive component and $\mathbf{v}_{i,\text{rep}}[t]$ is the velocity vector generated by the adaptive inter-agent repulsion component.

However, in contrast to the original strategy, this newly proposed strategy endows agents 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 neighbours. These received values are compared to an agent's own values and the most recent target position is used as a point of attraction, $\mathbf{p}$ (see Algorithm 4). Using this update algorithm, agents are able to compare information that is directly sensed from the environment with information received from its neighbours and choose which set of information to exploit.

Using this updated value of $\mathbf{p}$, the attractive velocity component of an agent is calculated using the same equation presented in the Chapter 3.

$$\mathbf{v}_{i,\text{att}}[t+1] = \omega \mathbf{v}_i[t] + cr(\mathbf{p}[t+1] - \mathbf{x}_i[t+1]). \quad (4.2)$$

The adaptive repulsion behavioural component of the agents is left unchanged from the previous chapter. As previously explained, the use of this behaviour promotes exploration of the environment when agents are not actively moving towards a target. The tracking agents also use the same interconnecting k -nearest neighbour communications network. Using this adjustable network, it was observed that a swarm can be made to favour area exploration or information exploitation by changing the value of k . As such the addition of memory is another parameter that can be used to tune the EED balance. This memory-augmented search and track strategy is summarised by Algorithm 5.

Algorithm 4 : 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

```

Algorithm 5 : Memory-Enabled Search and Track 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  $i \in [1, N]$  do
4:     Set  $\mathbf{p}$  using Algorithm 4
5:     Calculate  $\mathbf{v}_{\text{att},i}$  using Eq. (3.2)
6:     Calculate  $\mathbf{v}_{\text{rep},i}$  using Algorithm 3
7:      $\mathbf{v}_i[t] \leftarrow \mathbf{v}_{\text{att},i}[t] + \mathbf{v}_{\text{rep},i}[t]$ 
8:      $\mathbf{v}_i[t] \leftarrow (v_{\text{max}}/v_i[t]) \cdot \mathbf{v}_i[t]$ 
9:      $\mathbf{x}_i[t + 1] \leftarrow \mathbf{x}_i[t] + \mathbf{v}_i[t]$ 
10:   end for
11:    $t \leftarrow t + 1$ 
12: end while

```

4.4 Simulation Methods

As was the case previously, the employed targets do not emit a gradient field and agents are only able to detect a target's location when it lies within the target's radius. The use of such targets eliminates the possibility of agents following gradient decent methods to exploit a target's positional information. The same problem statement and simulation setup presented in Chapter 3 is also used. It should also be noted that for the experiments carried out with heterogeneous swarms, only virtual simulations were carried out.

In addition to the non-evasive target movement policy established earlier, a system's behaviour in the presence of evasive targets is compared and contrasted. While using the evasive policy, the targets initially follow the non-evasive policy, previously presented in Chapter 3, until it makes contact with an agent. Upon making contact with an agent (i.e. when an agent falls within the target's radius), the target attempts to move away from the agent by generating a velocity vector by using the repulsion equation presented in Equation (3.3), taking all agents within its radius as repulsion neighbors. After encountering agents for t_{limit} consecutive iterations, the target travels in a straight line for t_{evade} time-steps to attempt to outrun its pursuers. After t_{evade} expires, the target will revert to the non-evasive movement policy until it makes contact with an agent again. Like in the non-evasive study performed in the previous chapter, repeated simulations using the same settings and conditions yielded results that varied by $< 1\%$. Figures with the uncertainty ranges can be found in Appendix C.

4.5 Simulation Results

4.5.1 Homogeneous Swarms

Impact of Degree of Network Connectivity

Single Target Tracking While tracking both evasive and non-evasive targets, high target velocities resulted in a reduced collective tracking performance regardless of the level of connectivity, as seen in Fig. 4.2. Given that the swarming agents' velocities were not increased to match that of the target, increasing a target's velocity allows it to more easily outrun its pursuers, hence explaining the drop in the system's overall tracking performance. This is similar to the results obtained when tracking non-evasive targets seen in the previous chapter. Also, as can be expected, the tracking performances of systems pursuing evasive targets are worse than those pursuing non-evasive targets. Again, this is because an evasive targets makes its movements to avoid contact with the tracking agents, thereby making this a more challenging task.

From Fig. 4.2, it can also be observed that there exists an optimal degree of connectivity, $k = k^*$, at which the tracking performance is maximised when evasive target tracking is carried out. This is consistent with the results seen when tracking non-evasive targets, previously discussed in Chapter 3. Like in the non-evasive case, this optimum only appears when the target travels faster than the individual agents. This is because at low target velocities, it is not necessary for the system to regulate its level of exploration and exploitation to effectively track the target; the swarm can remain constantly engaged

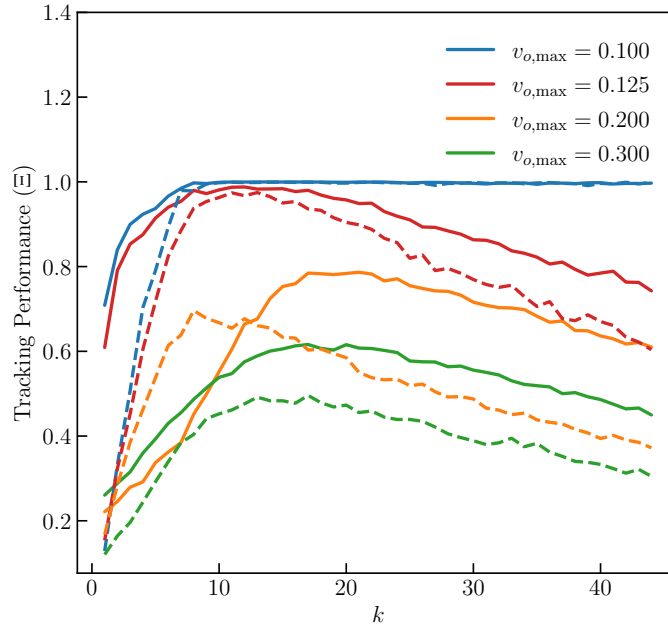


FIGURE 4.2: Non-evasive target (solid line) and evasive target (dashed line) tracking performance with different maximum target speeds, $v_{o,max}$. Maximum agent's velocity set at $v_{a,max} = 0.1$ and memory length of $t_{mem} = 20$ was given to the agents.

with the target and track it through performing purely exploitative actions. In contrast, when the target travels faster than the agents, the swarm needs to be able to disengage with the target to carry out more exploratory actions. This is especially true after being outrun by the target as the agents need to be able to re-explore the search space to re-acquire the target's location.

Like the non-evasive case, increasing the level of connectivity, k , when tracking an evasive target serves to increase the level of exploitation carried out by the system. This is illustrated in Fig. 4.3 that show increasing k also increases the engagement ratio, Θ . However, in contrast to the non-evasive tracking case, the optimal EED balance when tracking an evasive target favours higher levels of exploration. This is characterised by lower k^* values for the evasive target tracking case shown in Fig. 4.2. The lower k occurs because when a target moves in an evasive fashion, it is able to outrun its pursuing agents more frequently. As such, a system needs to carry out a higher level of exploratory activity to locate the target; the performances of systems tracking evasive targets are more exploration-limited than those tracking non-evasive targets. Using the proposed strategy, such high levels of exploration are achieved using lower levels of connectivity, resulting in k^* occurring at a lower value.

These results further stress the critical need to balance the amount of exploration and exploitation carried out by the swarm according to the task assigned. Doing so allows it to regulate its EED and carry out the exploratory and exploitative actions necessary to optimise a system's performance for a given task. This changing balance will be

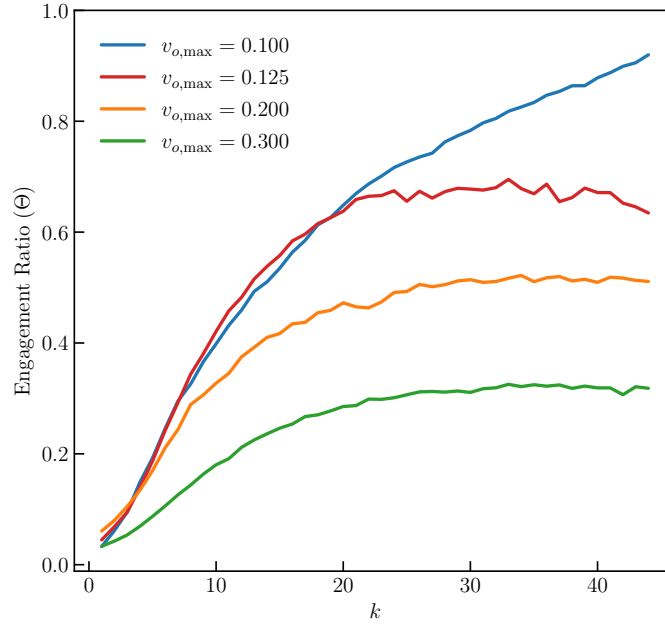


FIGURE 4.3: Swarm system engagement, Θ , with a single evasive target traveling at different maximum speeds, $v_{o,max}$. The maximum agent's velocity was set at $v_{a,max} = 0.1$ and memory length was set at $t_{mem} = 20$.

further discussed in Section 4.5.1.

Multiple Target Tracking Similar to single target tracking, the swarm has a lower k^* when tracking evasive targets compared to when tracking non-evasive targets (see Fig. 4.4). Again, this is due to the swarm requiring higher levels of exploration when pursuing evasive targets.

Also seen in Fig. 4.4 is a reduction in k^* when tracking multiple targets. This holds true for both evasive and non-evasive targets and is because more exploration is required to acquire information about the different targets' location. In addition, the greater number of targets in the search space also naturally results in a higher number of target encounters. These additional target encounters serve to increase the engagement ratio of the system, and hence pushing the EED balance in favour of exploitation. To maintain the optimum balance between exploration and exploitation, more area exploration must be carried out. These higher levels of exploration are achieved at lower levels of connectivity, resulting in a lower k^* when tracking multiple targets.

In addition to the reduction in k^* , the system's tracking performance is also shown to decrease when tracking multiple targets. This is because tracking multiple targets simultaneously is an inherently harder task than tracking a single target. Given that no changes are made to make the swarming agents more mobile, this reduction in tracking performance should be expected when more targets are added to the area of operation.

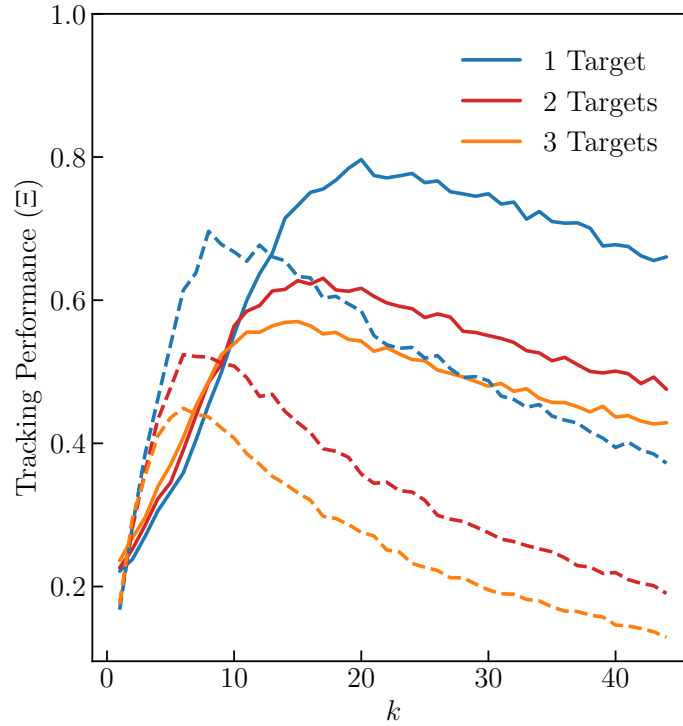


FIGURE 4.4: Multiple non-evasive (solid lines) and evasive (dashed lines) target tracking performance (maximum agent's velocity $v_{a,\max} = 0.1$, memory length of $t_{\text{mem}} = 20$, and maximum targets' velocity $v_{o,\max} = 0.2$).

Impact of Agent Memory Length

When attempting to track a non-evasive target in the previous chapter, a memory-less strategy was employed to prevent agents from exploiting outdated information. This is justified by Fig. 4.5 as it can be seen that when tracking a non-evasive target, the addition of memory results in a small decrease in the system's overall tracking performance. This performance degradation is caused by agents aggregating in a location where the target is no longer present.

However, Fig. 4.5 also shows the critical nature of short-term agent memory when tracking an evasive target. When an agent's memory length is set to $t_{\text{mem}} = 0$, the system is unable to effectively track the evasive target. As the memory length is gradually increased from a memory-less system, the tracking performance starts to improve rapidly until an optimum memory length is attained before seeing a degradation in performance. This is because the use of agent-based memory generates a persistent point of attraction based on a target's last known position, giving the agents the ability to aggregate at that point. While the swarm may periodically encounter the target without the use of memory, its agents will tend to repel each other when the target changes direction and agents are no longer able to detect the target. This occurs until the agents reach their static equilibrium positions. Given that the agents travel slower than the

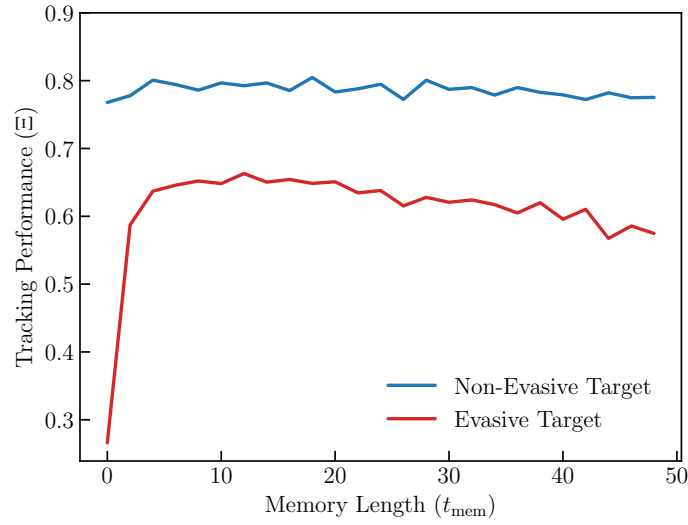


FIGURE 4.5: Single evasive and non-evasive target tracking performance with different implemented agent-based memory lengths (target speed $v_{o,\max} = 0.2$). Tracking agents are connected in a dynamic $k = 20$ network.

target, without the use of memory, the system will be unable to aggregate around the target, preventing it from achieving its goal of target tracking. This point is illustrated in Fig. 4.6 that shows that the system is unable to engage with an evasive targets with very low memory lengths.

Balancing Network Connectivity and Memory

When comparing Figs. 4.2 and 4.5, it can be seen that after the optimal memory length has been attained, changing the degree of connectivity impacts the MRS' tracking performance more compared to the effects attained by altering the memory length. However, Fig. 4.7 shows that changing the length of the memory present in each agent also affects the optimal degree of connectivity required to maximise the tracking performance. The figure shows that increasing agents' memory length tends to decrease the optimal degree of connectivity, k^* . This is caused by an agent's tendency to utilise target location information for a longer period of time with longer memory lengths, thus performing higher amounts of exploitation. This is congruent with the findings by Nauta et al. (2020), who showed that stronger memory effects favour more exploitative actions in a foraging scenario. Therefore, to maximise its tracking performance while using increased memory lengths, a swarm needs to compensate for this increase in exploitation by performing more exploratory actions. Given that the memory length was fixed in this set of experiments, this higher level of exploration can only be attained at lower levels of network connectivity, effectively resulting in a reduced k^* value.

These obtained results point to an optimal balance between exploration and exploitation carried out by the swarm to maximise its tracking response. This is demonstrated in Fig. 4.8, which shows a clear optimum level of engagement, Θ^* , that maximises the

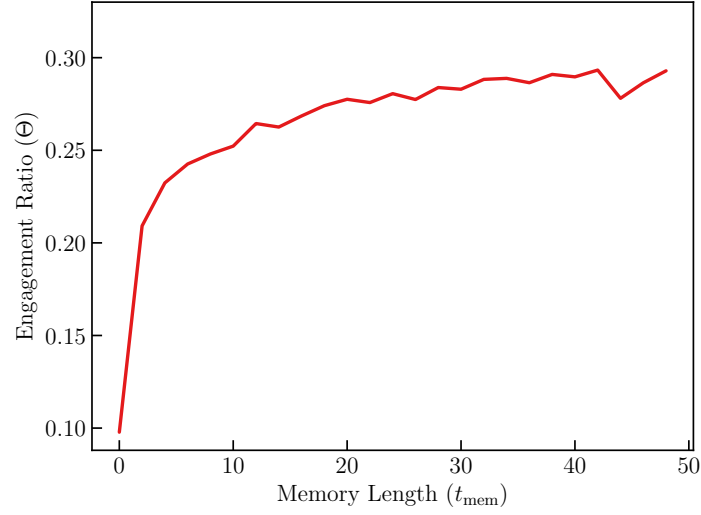


FIGURE 4.6: Evasive target system engagement with varying memory lengths. Target moved at a speed of $v_{o,\max} = 0.2$. Agents were connected in a dynamic $k = 20$ network.

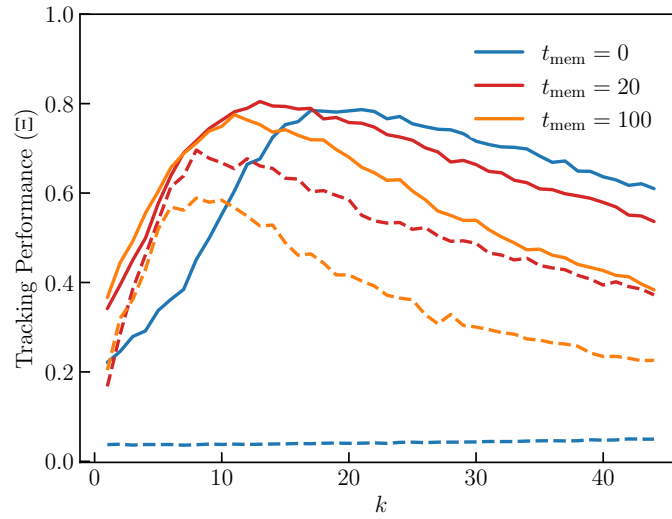


FIGURE 4.7: Single non-evasive (solid line) and an evasive (dashed line) target tracking performance while varying the length of the implemented agent-based memory and degree of network connectivity.

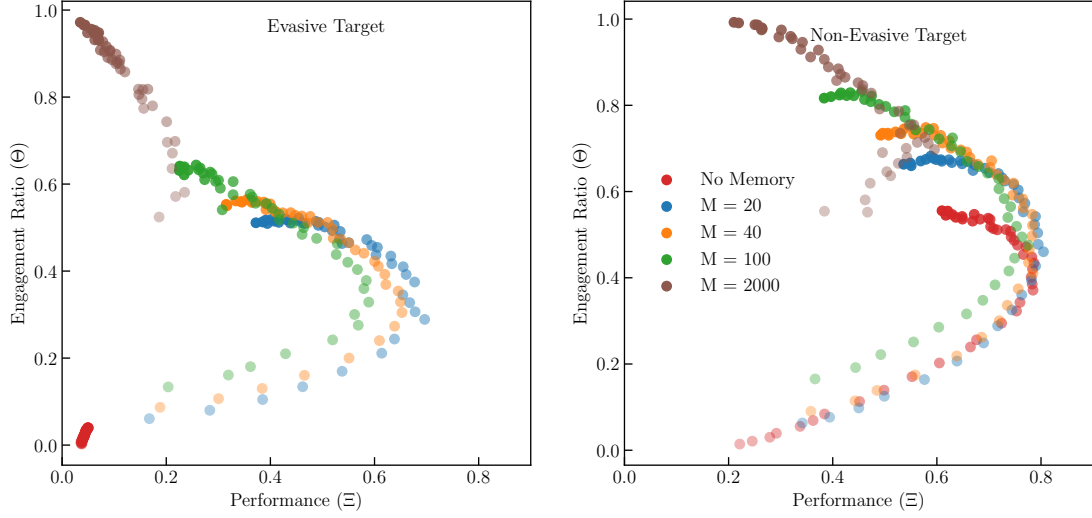


FIGURE 4.8: Engagement-Tracking plots of a swarm with different t_{mem} and k tracking an evasive (left) and a non-evasive (right) target. Darker shaded points indicate swarms using networks with higher values of k .

MRS' tracking performance. The plots also show that the ideal engagement ratio maximizing evasive target tracking performance is lower than that for a non-evasive target, $\Theta^* \sim 0.3$ for the evasive case compared to $\Theta^* \sim 0.5$ for the non-evasive case. This supports the conclusion that the tracking of evasive targets require higher amounts of exploration. Not only do these results highlight the presence of the optimal balance between exploration and exploitation, they also reveal the important fact that this optimum varies based on the task presented to the swarm.

Furthermore, Fig. 4.8 shows that excessive memory lengths ($t_{\text{mem}} = 2000$) degrade the tracking performance of the swarm when pursuing both evasive and non-evasive targets, especially in the non-evasive target tracking case. The plot also highlights the necessity of the inclusion of agent-based memory when tracking evasive targets as the system without memory is outperformed by all the other systems.

4.5.2 Strategy Comparisons

Having ascertained its tracking performance, the newly developed strategy with memory is compared to three other benchmark strategies. These benchmark strategies are: (1) $k = 0$ with memory—corresponding to the case where there is no inter-agent communication and cooperation, (2) $k = 10$ without memory—corresponding to the strategy introduced in the previous section, (3) $k = 49$ with memory—corresponding to the case where a global communications network is used, and (4) the standard PSO algorithm proposed by Kennedy and Eberhart, 1995. These results were obtained by simulating the system tracking a single fast-moving evasive target. The simulations were repeated 20 times for each of the four different strategies with different random seeds.

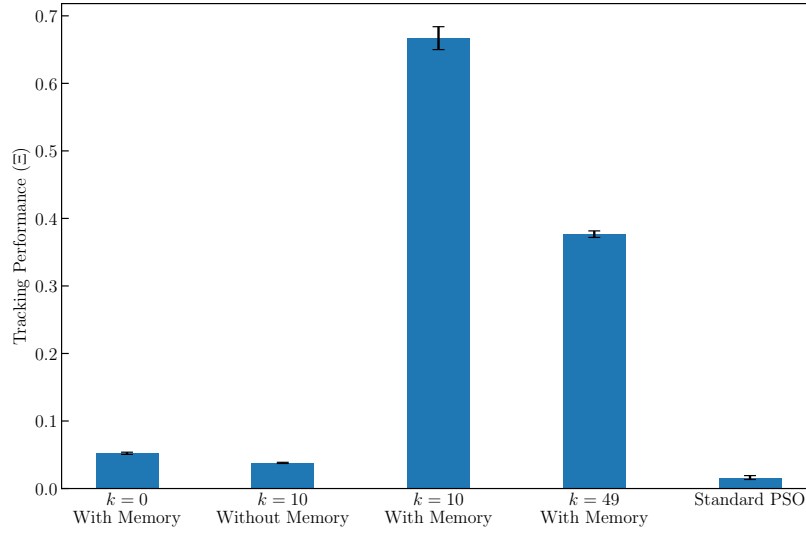


FIGURE 4.9: Tracking performance of swarms using the proposed search and track strategy with (1) a $k = 0$ network with memory, (2) a memory-less $k = 10$ network, (3) a $k = 10$ network with memory, and (4) a $k = 49$ network with memory, as well as a swarm using the standard PSO algorithm. The target moved at a maximum speed of $v_{o,\max} = 0.2$ while the pursuing agents travelled at a maximum speed of $v_{a,\max} = 0.1$.

Figure 4.9 shows that by using the newly proposed strategy with memory, a multi-agent system is indeed able to demonstrate a higher tracking performance when its level of connectivity has been optimised. This also further emphasises the critical role that memory plays in a swarm tracking an evasive target, as a memory-less swarm is shown to be unable to track the target. The figure also shows the ability of the system to outperform the standard PSO algorithm due to the strategy's ability to modulate its level of exploration and exploitation thanks to its lower level of connectivity and use of a short-term agent-based memory.

4.5.3 Heterogeneous Swarms

Tracking Performance

As can be seen in Fig. 4.10, increasing the proportion of fast agents results in an increase in the system's tracking performance, regardless of the number of targets present. Fig. 4.10a also reveals that these increases are proportional to the number of fast agents integrated into the system. Being more mobile, these fast agents are better able to keep up with the target, thereby directly improving the system's performance. In addition to the effect of substituting in fast-moving agents, Fig. 4.10 also unsurprisingly shows that fewer targets to be tracked results in better system performances. Again, this is because tracking fewer targets is a less demanding task for the MRS; simultaneous tracking of multiple targets does not need to be carried out.

It is worth highlighting that these increases in performance are obtained without modifying the original strategy and is similar to the results seen in the non-evasive case.

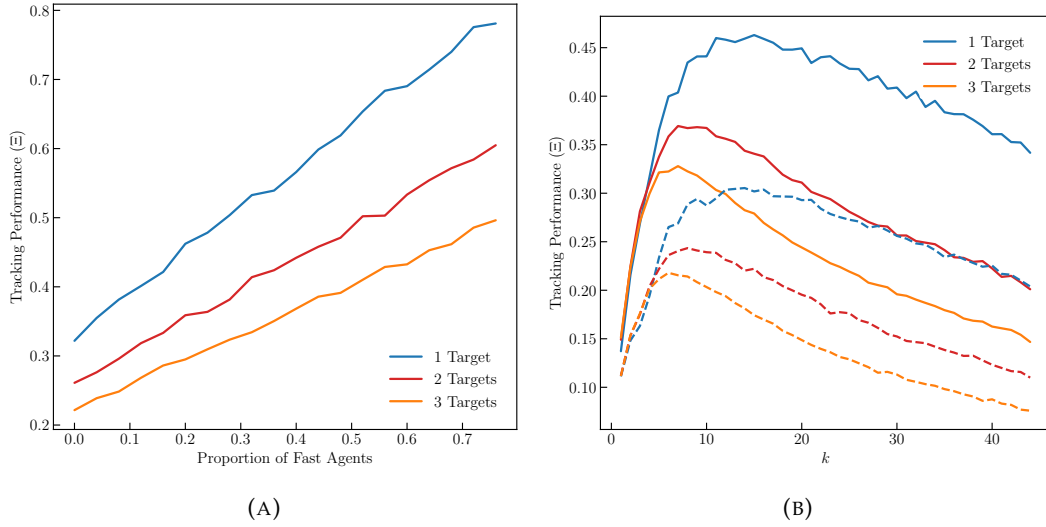


FIGURE 4.10: Tracking performance of a heterogeneous swarm tracking evasive multiple targets using a memory length of $t_{\text{mem}} = 20$. Simulations for (A) were performed using a network connectivity of $k = 12$ for different proportions of fast agents. Simulations for (B) were performed using 15 fast agents and 35 slow agents for different levels of connectivity.

Qualitative observations made from the simulations reveal the reasons why no modifications are needed. As illustrated in Fig. 4.11, while the faster agents may flock together when actively trying to engage with the target, they are eventually able to disperse and distribute themselves across the swarm after losing track of the target. This is thanks to the adaptive inter-agent repulsion component in the search and track strategy, allowing the agents to have different repulsion strengths, thereby facilitating this distribution and increasing the chances of a fast agent encountering a target.

Exploration-Exploitation Dynamics

In addition to the improved tracking performances when heterogeneous swarms are employed, Fig. 4.12a also shows a slight decrease in the system's k^* when faster moving agents are introduced, albeit less apparent than the decrease seen in the non-evasive case (See Chapter 3). While this reduction in k^* may seem to suggest that the EED of heterogeneous systems lies more in favour of exploration, Fig. 4.12b shows that this balance in fact lies more in favour of exploitation.

The reason for the reduction in k^* can be seen in Fig. 4.13. As the proportion of fast agents in the system increases, so too does the system's engagement ratio, Θ . This increase in the system's engagement ratio is again due to the faster agent's greater level mobility. As previously stated, since the faster agents are better able to keep up with the targets, they directly increase the swarm's tracking capability. Due to the longer time spent tracking the target, the agents are also able to disseminate more information about the targets' locations across the swarm, allowing neighboring agents to remain engaged with a target for longer and close in on it, thereby shifting the system's EED further in favour of exploitation. Should none of the system parameters be changed, the increased tendency to carry out exploitative actions due to the substitution of faster

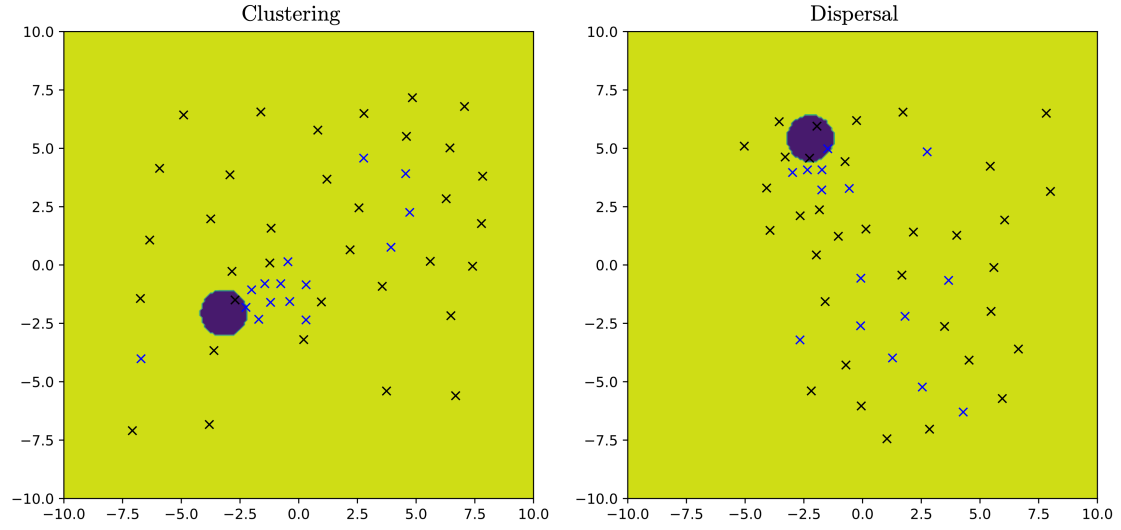


FIGURE 4.11: Heterogeneous swarm comprised of 35 slow-moving agents (black crosses) and 15 fast-moving agents (blue crosses) tracking an evasive target (dark blue disc). (Left) Fast agents clustering while pursuing a target; (Right) After losing track of the target, fast agents distribute themselves around the system due to the variable inter-agent repulsion.

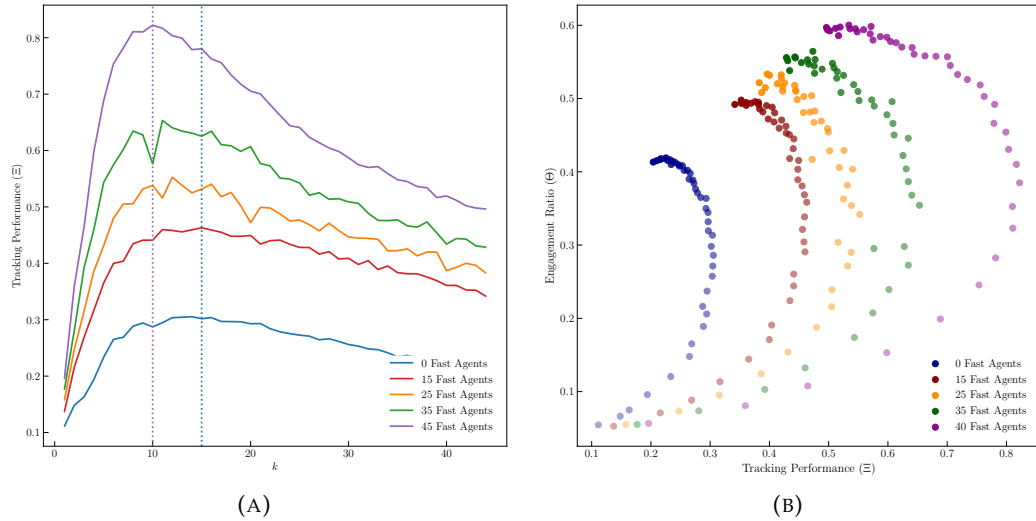


FIGURE 4.12: Swarms comprised of 50 agents with different agent compositions of fast and slow tracking a single fast-moving evasive target. (A) Tracking performance of the system while operating with different levels of connectivity k with the optimum k decreasing as the number of fast agents present increases. (B) Engagement-Tracking plots where darker shaded points indicate swarms communicating through networks with higher values of k .

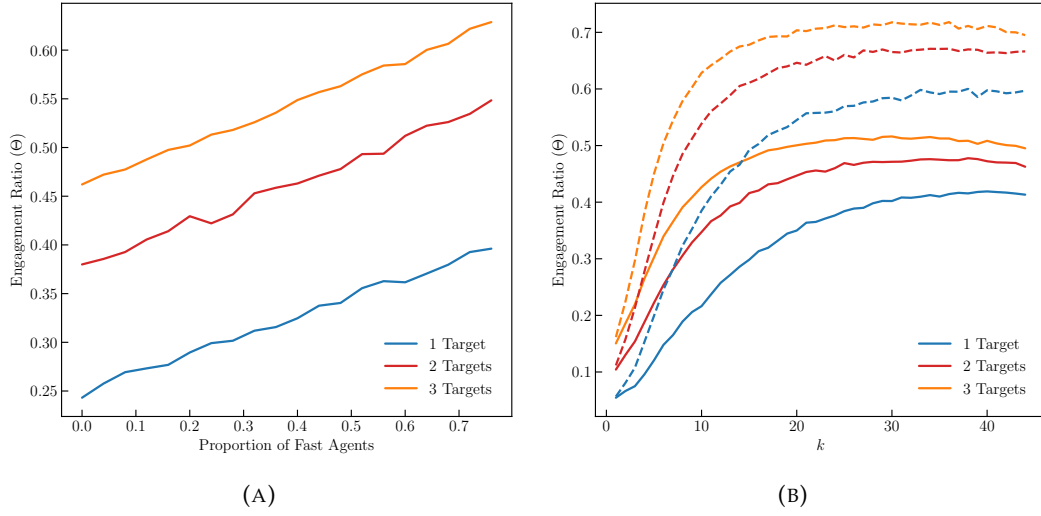


FIGURE 4.13: Engagement ratio of a heterogeneous swarm tracking evasive multiple targets using a memory length of $t_{\text{mem}} = 20$. Simulations for (A) were performed using a network connectivity of $k = 12$ for different proportions of fast agents. Simulations for (B) were performed using 15 fast agents and 35 slow agents for different levels of connectivity.

moving agents will cause a system's performance to become more exploration-limited. As such, to reattain the optimal EED balance, the system needs to carry out more exploration, a task achieved by lowering its level of communication, thereby resulting in a lower k^* .

However, despite the need to lower k to increase the amount of exploration carried out by heterogeneous systems, Fig. 4.12b shows that the optimum EED balance shifts in favour of exploitation. This suggests that the primary effect of substituting in fast agents into a swarm is the increase of both the system's capability and propensity to exploit target information. Faster agents are better able to follow the target, thereby increasing the capability of the swarm to exploit information, and as a result, are able to disseminate target information for longer periods of time, thereby increasing exploitative tendencies. To balance the increased amount of exploitation promoted by the faster agents, the system needs to carry out more exploration. As more exploration is carried out at lower levels of connectivity, the optimum connectivity degree, k^* , decreases when faster agents are substituted into the system.

4.5.4 Heterogeneous Strategies

The usage of different agents within a swarm opens up the possibility of developing differentiated strategies to further take advantage of the heterogeneous character of the system. The usage of such differentiated strategies would potentially allow for higher overall tracking performance. Here, the usage of heterogeneous agent memory and communications strategies is explored.

Since it was previously determined swarms of different compositions had different optimal EED balances, simulations were carried out using homogeneous swarms while

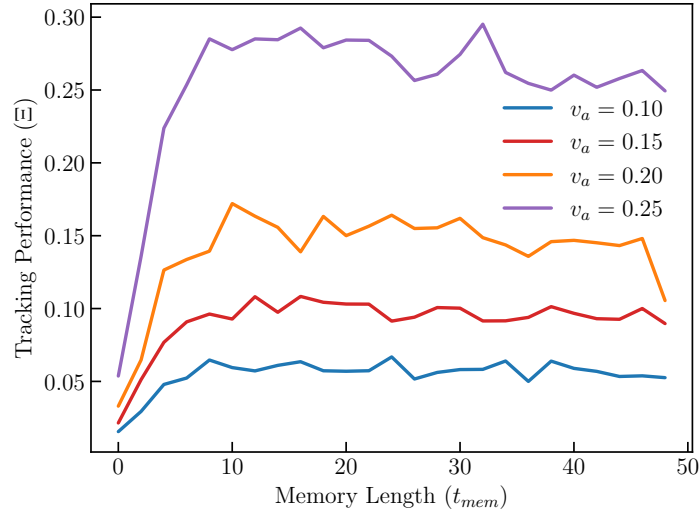


FIGURE 4.14: Evasive tracking performance of homogeneous swarms with different $v_{a, \max}$, and memory lengths, t_{mem} . Simulations were run at $k = 14$ and $v_{o, \max} = 0.3$

utilising different memory lengths, as well as levels of connectivity. To ascertain the possibility of using a heterogeneous memory strategy. However, as shown in Fig. 4.14, the optimum memory length does not appear to change with the maximum speed of the agents. This therefore suggests that the use of a heterogeneous strategy involving the use of different memory lengths will not improve the tracking performance of a swarm.

As previously demonstrated, substituting in faster moving agents serves to reduce the optimal level of connectivity for the swarm. In addition, other studies have shown that the optimal level of connectivity changes with the speed of the driving stimulus (Mateo et al., 2019). As such, to explore the possibility of using a heterogeneous strategy using different levels of connection, evasive target tracking simulations were carried out. In this set of simulations, the level of connectivity of the fast agents (k_f) was set to be independent from the level of connectivity of the slow agents (k_s).

From the simulations performed, an optimum combination of levels of connectivity to maximise the tracking performance is found (see Fig. 4.15). However, this optimum occurs when $k_s = k_f = 10$, i.e., when a homogeneous connectivity strategy is used. None of the other cases tested suggest that employing a differentiated strategy would further improve a swarm's tracking performance. These results hint that further attempts to improve the tracking performance of heterogeneous MRS swarms should integrate other behavioural factors: e.g., differentiated task allocation based on an agent's placement, the utilisation of target movement pattern predictions, etc.

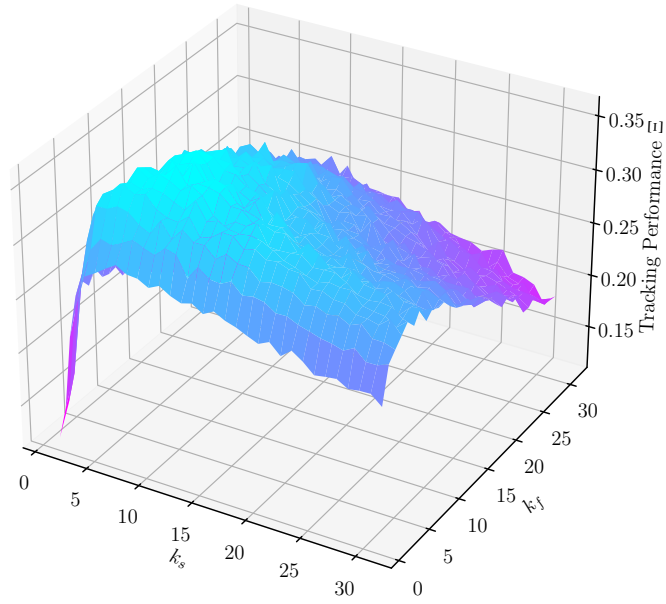


FIGURE 4.15: Single evasive target tracking performance of a heterogeneous comprised of 30 slow agents and 10 fast agents with slow and fast agents, each having different levels of connectivity.

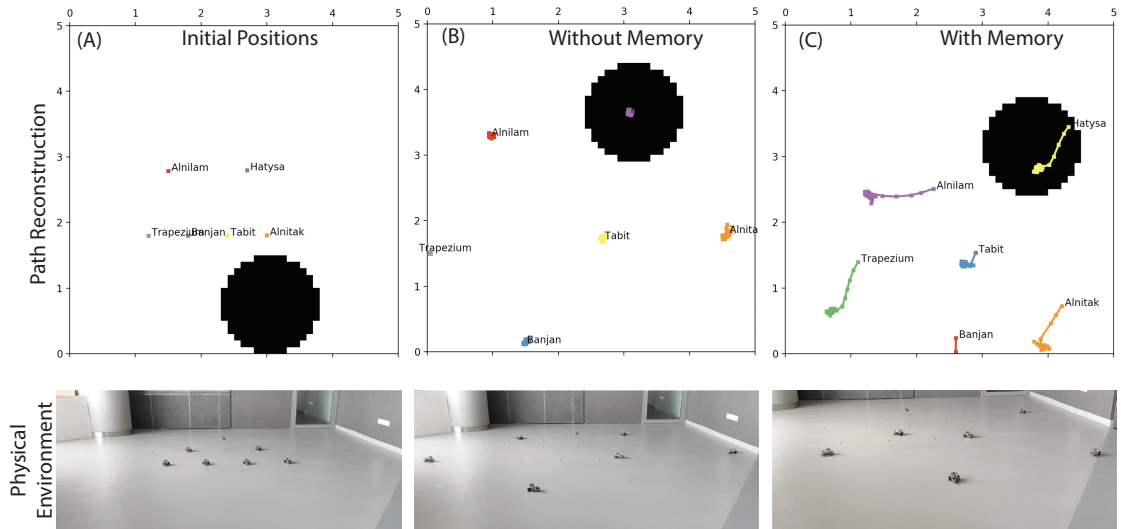


FIGURE 4.16: (Top) Reconstruction of target (black circle) and robot positions and robot paths (coloured traces) as observed through our monitoring system. (Bottom) Physical experiments using a connectivity of $k = 5$ and adaptive repulsion. (A) Robots initialised at the start of the experiments. (B) Robots without agent-based memory unable to respond to a target's presence. (C) Robots with agent-based memory moving towards target after being found.

4.6 Robotic Experiments

4.6.1 Experiment Conditions

To test the validity of the proposed evasive target tracking strategy for a homogeneous system, a swarm of 6 ORION robots (previously used for the non-evasive target tracking task in Chapter 3) is utilised. At the start of all tests, these 6 robots are arranged in the middle of an open search area (see Fig. 4.16). The robots are tasked with tracking a virtual evasive target, traveling at an average speed of 5 cm/s, following the same evasive movement policy previously described in Sec. 4.4 with $r = 0.75$ m. The target's location is determined by a central computer that calculates its next position based on the current locations of all of robots. It should again be noted that while the target traveled at a slower speed than the robots, the target has infinite maneuverability compared to the finite maneuverability of the robots. As such, even if the individual robots can move faster than the target in a straight line, they effectively respond much slower due to their low maneuverability.

All experiments are run with an all-to-all connectivity ($k = 5$), in an attempt to make the response of the swarm more apparent. Two sets of tests were run, one with agent-based memory length of $t_{\text{mem}} = 15$ s, and another without any (i.e., $t_{\text{mem}} = 0$ s). Each set of tests is comprised of 10 runs, with each run lasting for approximately 2 minutes to minimise the directional drift experienced by the robots and allow for the assumption that the robots are able to accurately self-localise.

4.6.2 Experimental Results

TABLE 4.1: Swarm tracking performance results with and without agent-based memory obtained from the robotic units.

	Without Memory		With Memory	
	39.0	31.8	48.8	63.8
	47.1	39.9	46.6	44.0
	37.6	40.5	64.4	37.0
	32.9	37.1	40.1	58.4
	30.8	35.1	48.1	42.8
Averaged	37.2 ± 4.6		49.4 ± 9.2	

The robots' locations are communicated to a central computer to enable the reconstruction and visualisation of their paths and responses to the presence of a target seen in Fig. 4.16. The figure shows that without memory, the robots are unable to respond to the presence of the target and tend to remain in place even when the target had been encountered. In contrast, with agent-based memory, when an agent encountered the target, the swarm as a whole responds to the target's presences and the units turn to move towards it and aid in the tracking of the target. The improved response of swarm system to the presence of the target resulted in a significant increase in tracking performance ($p < 0.01$), with the swarm being better able to track the evasive target for a

longer period of time (Table 4.1). This is similar to the results obtained in the simulations, validating the hypothesis that agent-based memory is required in the tracking of a fast-moving evasive target.

While the physical experiments performed confirmed the necessity of the implementation of agent-based memory, the quantitative results obtained from the physical experiments did not replicate those obtained from the simulations. Similar to the non-evasive target tracking experiments, these discrepancies can be attributed to three reasons: (1) the small number of robots used for the simulations, (2) the short experiment duration relative to that of the simulations, and (3) the physical tests and the simulations were not performed at the same scale and density. Also as demonstrated by Cziráok, Stanley, and Vicsek, 1997, statistical characterization of a multi-agent system's order becomes more difficult at low swarm densities. This makes it harder to predict the behaviour of a system with low density within simulations, which could account for some of the discrepancies between the simulation and physical experiment results.

4.7 Discussion

In this chapter, the search and track strategy proposed in Chapter 3 was further developed for implementations of swarms tracking fast-moving evasive targets. This was done by giving agents access to a short-term memory, thereby allowing them to generate a longer lasting point of attraction. This in turn facilitated the conduct of exploitative activities despite agents not having a target within their detection range. However, it was also shown that caution should be taken to ensure that excessive memory lengths are not used as they result in the system over-exploiting target information, thereby leading to a degradation of tracking performance. The strategy was further validated with physical robotic experiments, demonstrating the critical nature of agent-based memory in the tracking of a highly maneuverable fast-moving target.

The results obtained through simulations also show that the optimum degree of connectivity is lower when a system carries out evasive target tracking compared to when it carries out non-evasive target tracking. This is because the former is a task that demands higher level of exploration since the target moves to avoid being detected by the agents. This higher level of exploration is achieved by lowering the system's level of connectivity, thereby reducing the optimum level of connectivity, k^* . A similar drop in k^* was also observed when comparing the multi-target tracking to single-target tracking. Again, this reduction in k^* was due to the system increasing its level of exploration to search for more targets, as well as to compensate for the additional levels of exploitation provoked by the higher probability of encountering a target.

Similar to the original memory-less strategy, it was also shown that the new strategy with memory is able to be used in heterogeneous swarms comprised of agents of different speeds. During its implementation, increasing the proportion of fast agents in the system is shown to result in better tracking performances. This observation can be attributed to the higher mobility of the faster agents that gives them the ability to better keep up with and track the targets. With the fast agents better able to follow a target, information on the targets' locations are disseminated for longer periods of time, leading to higher levels of engagement with the target. This suggests that the

addition of fast agents into a system increases a system's exploitation tendencies and capabilities, as well as shifts the overall EED balance of these types of heterogeneous systems in favour of exploitation. However, even though this optimum EED shifts in favour of exploitation, the additional amount of exploitative activity generated through the introduction of fast-moving agents must be tempered; simply inserting fast-agents into a system without modifications to the overall strategy and behaviour can cause the swarm to perform too much exploitation, thereby becoming a system that has its performance limited through a lack of exploration. As such, to maximise the performance of such heterogeneous systems, more exploration must be encouraged, a task achieved by reducing a system's level of connectivity. Therefore, the results also show that changing the composition of the swarm results in changes in the optimum degree of connectivity due to the changing optimal EED balance, with systems comprised of higher proportions of fast agents displaying lower k^* .

Complementing the work presented in the previous chapter, the results laid out in this chapter show another two parameters that need to be considered when finding the optimal EED balance of the system: the movement profile of the target (i.e. the target's evasive or non-evasive behaviour) and the system's level of heterogeneity. When accounting for the system's level of heterogeneity, both the ability and tendencies of the individual units to carry out exploration and exploitation must be considered in strategy development. For example, systems that employ agents that have higher exploitation capabilities and tendencies, as was the case here, must utilise strategies that can moderate the amount of exploitative actions carried out to ensure that the entire system is able to maintain the optimal EED balance.

Furthermore, the results in this chapter also indicate that in environments where pertinent information is scarce, efforts need to be made to continuously exploit what little information is available to the swarm. However, this is done at the risk of the exploited information becoming outdated. As such, a balance needs to be struck on how long a system should spend exploiting the same piece of information.

Chapter 5

Swarm Density

5.1 Introduction

One of the key design parameters that an MRS designer must consider is selecting the number of robotic units required to achieve a desired performance for a given task and environment. In many MAS, be it an engineered system or one occurring naturally, it has been shown that effective system performances emerging from collective behaviour requires a critical number of swarming agents. 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 its size. For a given task, several other aspects also influence the selection of the number of swarming agents, such as: (1) system scalability (i.e., the capacity of a system to continue functioning properly when the number of its components substantially varies) (Dorigo, Theraulaz, and Trianni, 2021), (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, Trease, and Arsie, 2019).

Current MRS sizes cover three orders of magnitude. These range from the 1024-unit Kilobot swarm by Rubenstein et al. (2014), to the 150 unit subCULTron system by Theinius et al. (2016), to the hunter drone swarm by Souza, Castillo, and Vidolov (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. However, in the vast majority of works, it is often unclear how this quantity is selected. Would a half-sized Kilobot swarm perform vastly differently from the full-sized Kilobot MRS? At the other end of the spectrum, the same question can be posed to system designers who utilise 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), 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 just discussed.

The content in this chapter is intended for publication in:
Kwa, H.L., Philippot, J., and Bouffanais, R. [In Review]. “Effect of Swarm Density on Collective Tracking Performance”. In: *Swarm Intelligence*.

One of the main parameters taken into consideration when sizing an MRS is the dimensions of the planned operating environment. This inextricably links the problem of MRS sizing to the density of agents. Indeed, many works hint at the existence of a minimum critical density required to yield effective swarm self-organisation. For example, individual Kilobot units 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. In both of these systems, a high swarm density is required due to their limited communication ranges. Beyond the constraints in communications or information exchange, a scaling law has been used to quantify the effectiveness of the BoB system when placed a dynamic area monitoring scenario 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 behaviour in simplistic physics models, such as the paradigmatic self-propelled particles (SPP) in Vicsek's model (Vicsek et al., 1995). Regardless of the amount of noise present in the system, it was shown that the agent density is key in ensuring the self-ordering process. Specifically, at low density, the SPP swarm exhibits a disordered state that undergoes a phase transition leading to an ordered phase, the phase where 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 super-linear performance increases when the system was enlarged, followed by a decrease in performance once the swarm exceeded a certain size. This performance degradation was attributed to the increasing effect of interference on the system's performance, resulting from the elevated amount of collision avoidance behavior carried out by the agents. Besides collective motion, the problem of a system's exploration-exploitation balance, also needs to be considered. In various computational optimisation tasks, such as virtual agents carrying out particle swarm optimisation, and in the operation of MRS, larger swarms have been shown to exhibit higher levels of exploratory activity (Shishika and Paley, 2019; Piotrowski, Napiorkowski, and Piotrowska, 2020). However, as seen in the previous two chapters, 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.

Besides performance concerns, the failure to account for a system's density during the development process may result in a mismatch between the behaviours and performance predicted in virtual models and those observed in an actual system deployed in a physical environment, thereby widening the simulation-reality gap (Bozhinoski and Birattari, 2021). This was evident in the results presented in Chapters 3 and 4. These results showed that while there were clear benefits to using the newly proposed strategies in physical systems, the final performances of the robot swarm were notably

poorer than those modelled by the simulations. In addition to the lack of modelling interactions between the robots and their physical environments, which also played a role in this discrepancy (Zhong et al., 2018), a large part of deviation between simulated and empirically obtained results can be attributed to the different densities used.

In light of this, the work presented in this chapter studies the effects of density on the tracking performance of an MRS using the strategies previously presented in Chapters 3 and 4. Given the importance of a swarm's exploration and exploitation balance, the influence of swarm density on this balance is also examined with the goal of showing how this density ultimately affects the overall system performance. The investigation reveals three density 'phases' that characterise the tracking ability of an MRS, namely: (1) a low density phase where the system is simply unable 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. Here, we identify the factors that affects the transition between these phases. Finally, the results indicate that while the swarm density is an important element in swarm design, the total number of agents used within an MRS must also be considered even when the swarm density is kept constant. It should be noted here that while there are also stochastic methods that attempt to control agent density, such as those using Markov processes (Açikmeşe and Bayard, 2012; Elamvazhuthi and Berman, 2019; Biswal, Elamvazhuthi, and Berman, 2021), these are not in the scope of this thesis. This work focuses mainly on strategies employing deterministic methods, in line with the strategies already presented.

5.2 Related Work

5.2.1 Swarm 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 occurs 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 and Engelbrecht, 2001; Piotrowski, Napiorkowski, and Piotrowska, 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 optimisation 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, De Masi, and Ferrante, 2019; Shishika and Paley, 2019; Jurt et al., 2022). Similar to the MAS used in optimisation and

those found in Nature, these increases in performance are attributed to the increased exploration carried out by the larger systems.

5.2.2 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) and Sung et al. (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 optimisation, it is often reported that increasing the number of agents beyond a certain point yields a marginal benefit to the system's accuracy and notably lengthens computational times (Roeva, Fidanova, and Paprzycki, 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, Trease, and Arsie (2019) combined this multi-agent performance strategy with a simple cost model to maximise 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, Carmichael, and Moses (2015) concluded that when agents were tasked with depositing gathered 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. This is in contrast to the shorter distances to be travelled should the robots be allowed to deposit their collected resources in multiple locations scattered around the environment. In Rosenfeld, Kaminka, and Kraus (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 (2018) 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 underutilised nor over-utilised, resulting in the optimal performance of the system.

5.2.3 Motility-Induced Phase Separation

The increased amount of physical interference between agents is more apparent in systems tasked with clustering around certain points, such as those carrying out target tracking and consensus tasks. As previously mentioned, high levels of clustering results in elevated 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 and 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, including the ones presented in this thesis, have incorporated inter-agent repulsion to prevent the excessive flocking of robots at a single location, thereby promoting area exploration (Zhang et al., 2019a; Dadgar, Couceiro, and Hamzeh, 2020).

5.3 Simulation Methods and Metrics

The same simulated operating environment introduced in the previous chapters is used for this investigation on swarm density. To vary the swarm density, the width of the operating space, L , and the number of agents in the swarm, N are both varied. When N is kept constant, L is changed such that $L \in [10^{0.6}, 10^{2.65}]$. Similarly, when L is kept constant, N is varied such that $N \in [20, 30, 40, 50]$.

Instead of the engagement ratio used previously, in this chapter, the exploration ratio, Λ , is used to study the overall EED balance of the system. This is simply defined as:

$$\Lambda = 1 - \Theta. \quad (5.1)$$

In place of the engagement ratio, the tracking performance (previously defined in Eqn. (3.5)) is used to quantify the level of *effective* exploitation; i.e., exploitation that contributes to tracking of the target. This is in contrast to the engagement ratio that includes ‘wasted’ exploitation, such as the exploitative activity carried out by agents clustering at a location where the target is no longer present.

In addition, a new metric is introduced for the study of agent density or swarm density. It should be noted that these two terms are used interchangeably. This is calculated as follows:

$$\rho = N/L^2. \quad (5.2)$$

Moreover, 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 A 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:

$$\rho_L = \frac{1}{NT} \sum_{t=1}^T \sum_{i=1}^N \frac{7}{\pi L_i^2}, \quad (5.3)$$

where L_i is the average distance from an agent to its six closest neighbours (see Appendix A for the justification of why six neighbours were used). 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.

5.4 Results

5.4.1 Effect of Swarm Density on Tracking Performance

To investigate the effects of swarm density on tracking performance, simulations were run with different swarm sizes, ranging from $N = 20$ to $N = 50$, tracking a non-evasive target. In Fig. 5.1, 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. It is worth stressing that in the model used in this set of simulations, the quality of information exchanges remain unaffected by the large inter-agent distances, i.e., the agents keep the same number k of topological neighbors 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 so-called transition phase can be traced to the ability of agents to effectively cluster around a point of attraction. This ability to

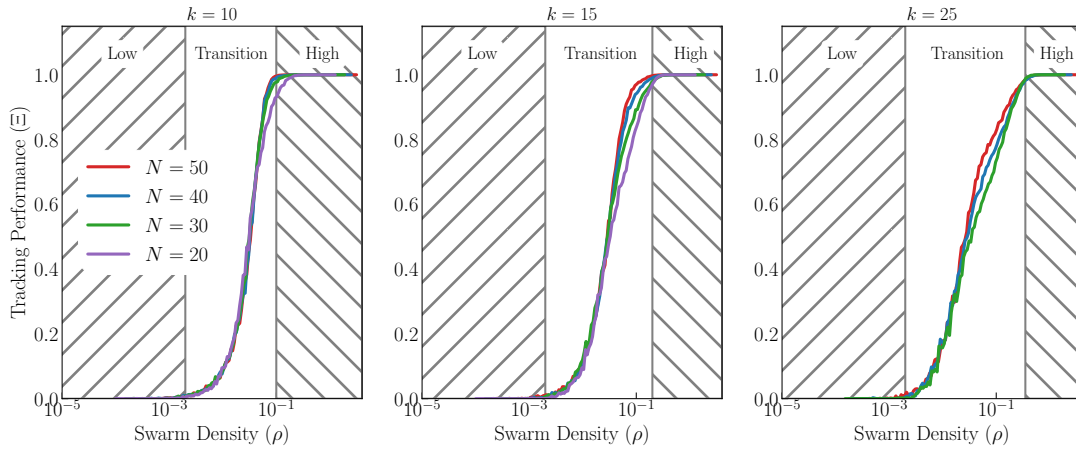


FIGURE 5.1: A swarm of $N = 20, 30, 40$, and 50 agents with varying levels of connectivity k and swarm density ρ tracking a fast-moving non-evasive target traveling at $v_{o,\max} = 0.15$. In the low density phase, the system is unable to track the target. In the high density phase, the system is able to track the target continuously without interruptions. Between these two phases, there is a transition phase in which the tracking performance of systems rapidly increases with swarm density, ρ .

cluster is revealed by the increase in density difference $\Delta\rho$ versus ρ (Fig. 5.2). The density at which the transition phase starts appears to be independent of N (Fig. 5.1). 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 outperform smaller ones, 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 will be explored in the subsequent sections. The effect of other factors on the system's behaviour in the transition phase can be found in Appendix B

5.4.2 Effect of Swarm Density on Exploration-Exploitation Dynamics

To determine the effects of changing a swarm's density on its EED, the system's level of exploration is calculated by calculating an *exploration ratio*, Λ , which is a system-level metric. It should be noted again that this Λ is the inverse of the exploitation ratio, Θ , introduced and used in the previous chapters. A system with a high Λ value will correspondingly have a low Θ , thereby indicating that the swarm is biased towards carrying out exploration. Conversely, low Λ values are associated with high Θ values and point to a shift of this bias towards exploitation. In addition to quantifying the performance of the swarm, the tracking performance Ξ can also be used to quantify the effectiveness of the swarm's exploitative actions; high levels of Ξ serve as an indicator

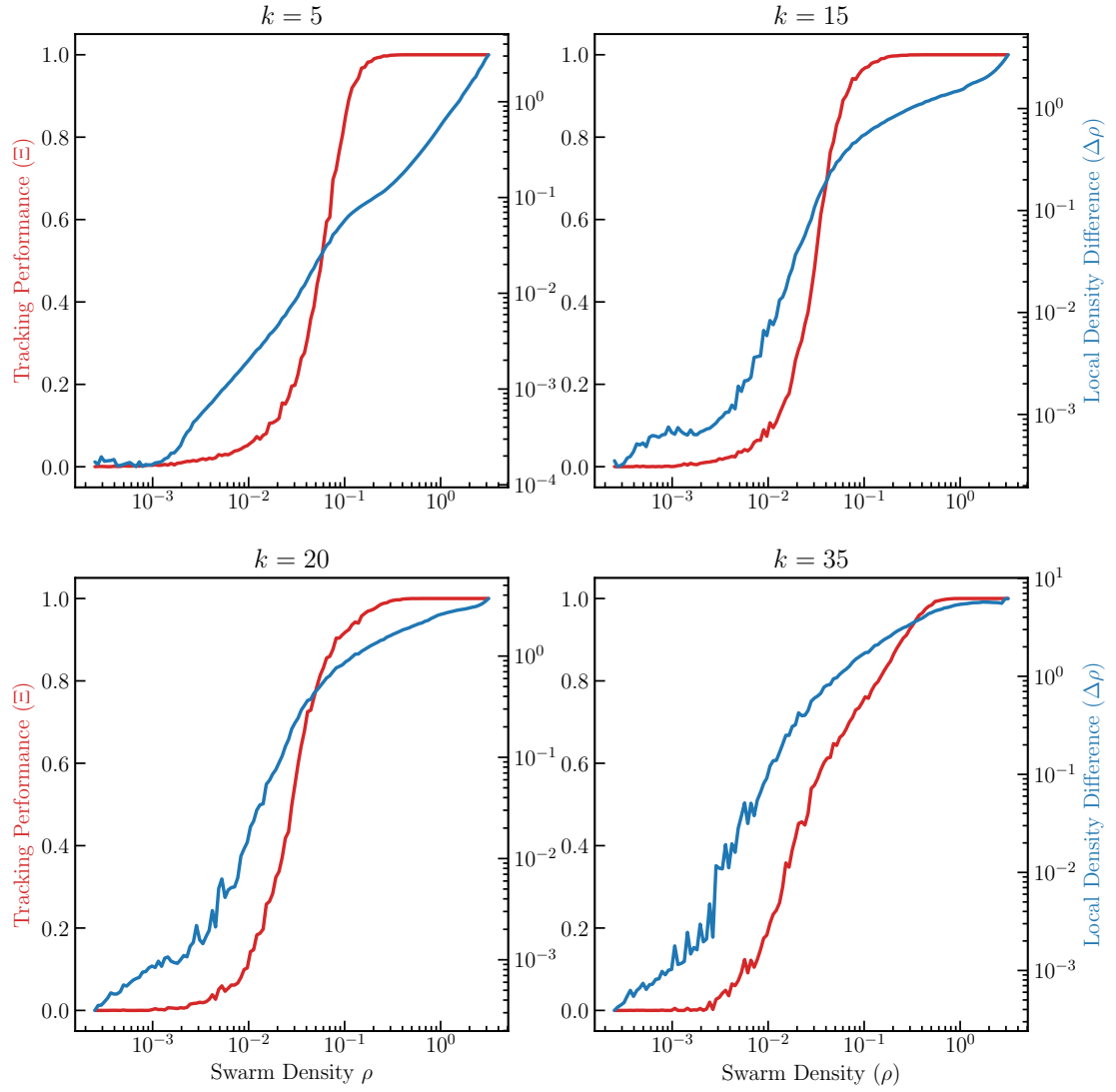


FIGURE 5.2: Density difference $\Delta\rho$ of a swarm comprised of $N = 50$ agents with various degrees of connectivity k , tracking a fast-moving non-evasive target traveling at $v_{o,\max} = 0.15$.

of a high level of effective exploitation. Note that in what follows, any optimum sought will be with respect to maximising Ξ .

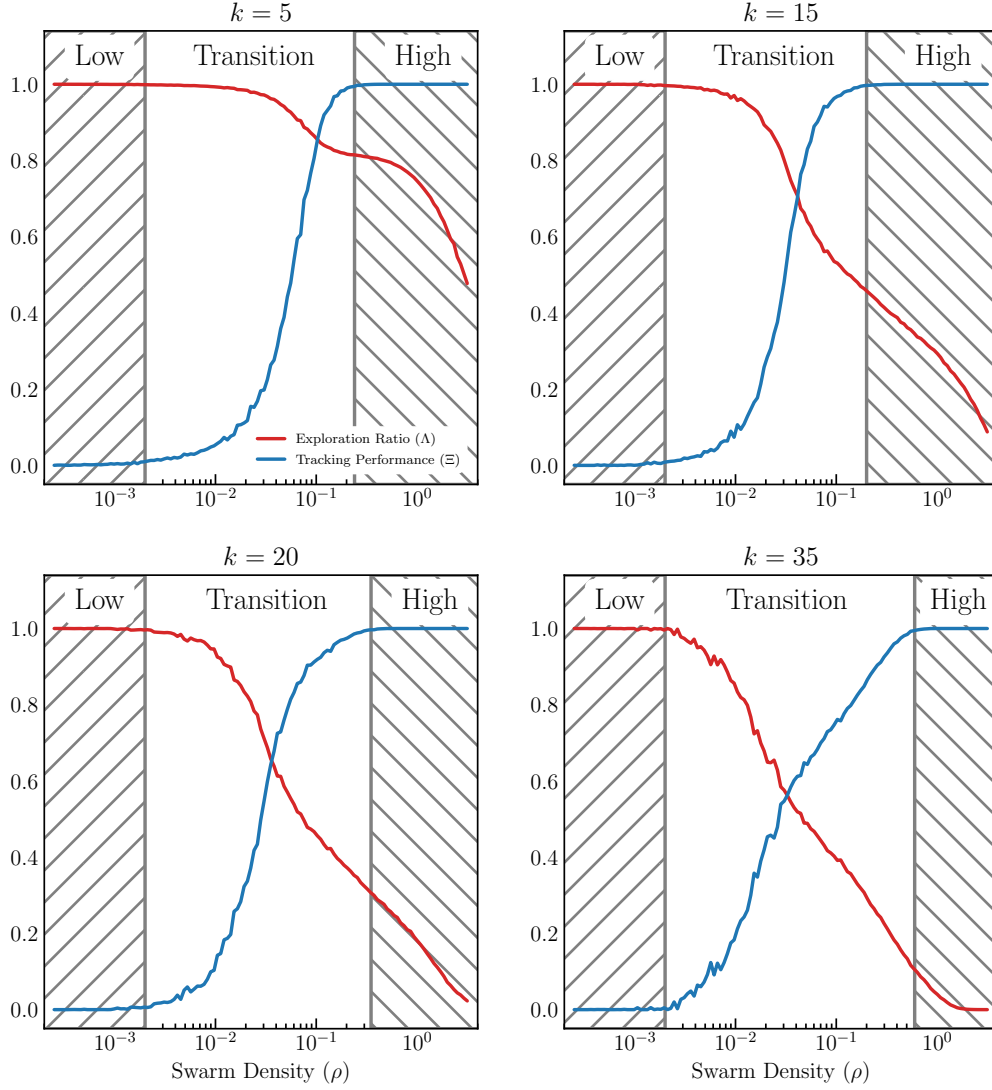


FIGURE 5.3: Tracking performance, Ξ , and exploration ratio, Λ , of a swarm of $N = 50$ agents using various levels of connectivity, k , while tracking a non-evasive target traveling at $v_{o,\max} = 0.15$.

The existence of a trade-off between exploration and exploitation for collective tasks has previously been reported in different MAS (Oliveira et al., 2017). However, a clear understanding of the underpinning of this trade-off is lacking. The work in this section serves as a quantitative analysis of this EED balance across the density spectrum. Starting with the low density phase identified in Fig. 5.1, the EED balance is fully skewed towards exploration regardless of the level of connectivity k , as evidenced by Fig. 5.3. At such low swarm density, even heavily favouring the conduct of exploitative actions through high levels of k is insufficient to yield minimal tracking of the target; the swarming strategy is plainly ineffective. As the system moves into the transition phase, which can be seen in Fig. 5.3, 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 increases, more resources become allocated to exploitation and the exploration ratio Λ decreases 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 in Fig. 5.3, a rapid decrease in Λ can be observed across the entire transition phase, almost to the point where no exploration is carried out at all. This should be compared to the $k = 5$ case, 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, Λ . 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, which has been explored in Chapter 2, where it has been shown that systems are able to carry out both exploration and exploitation simultaneously. Finally, when moving to the high density phase, which is exploration-limited, one is 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 Λ .

This rich and complex set of behaviours highlights an important fact: adaptivity, the ability of the system to modify its actions, in the MAS behaviour is thereby 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 the 'adaptivity' lever. At this stage, the next question arising is how can one 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 by the grey-shaded region in Fig. 5.4a that reveals the existence of a crossover in performance within the transition region. Specifically, at low density, a higher level of tracking is achieved with a higher connectivity, while at higher density (beyond the crossover), the converse is true.

To gain better insights into these phenomena, we turn to the variations of the exploration ratio Λ with ρ . Low degrees of connectivity, k , are associated with higher levels of exploration across the density spectrum (Fig. 5.4b). Indeed, a reduced k 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

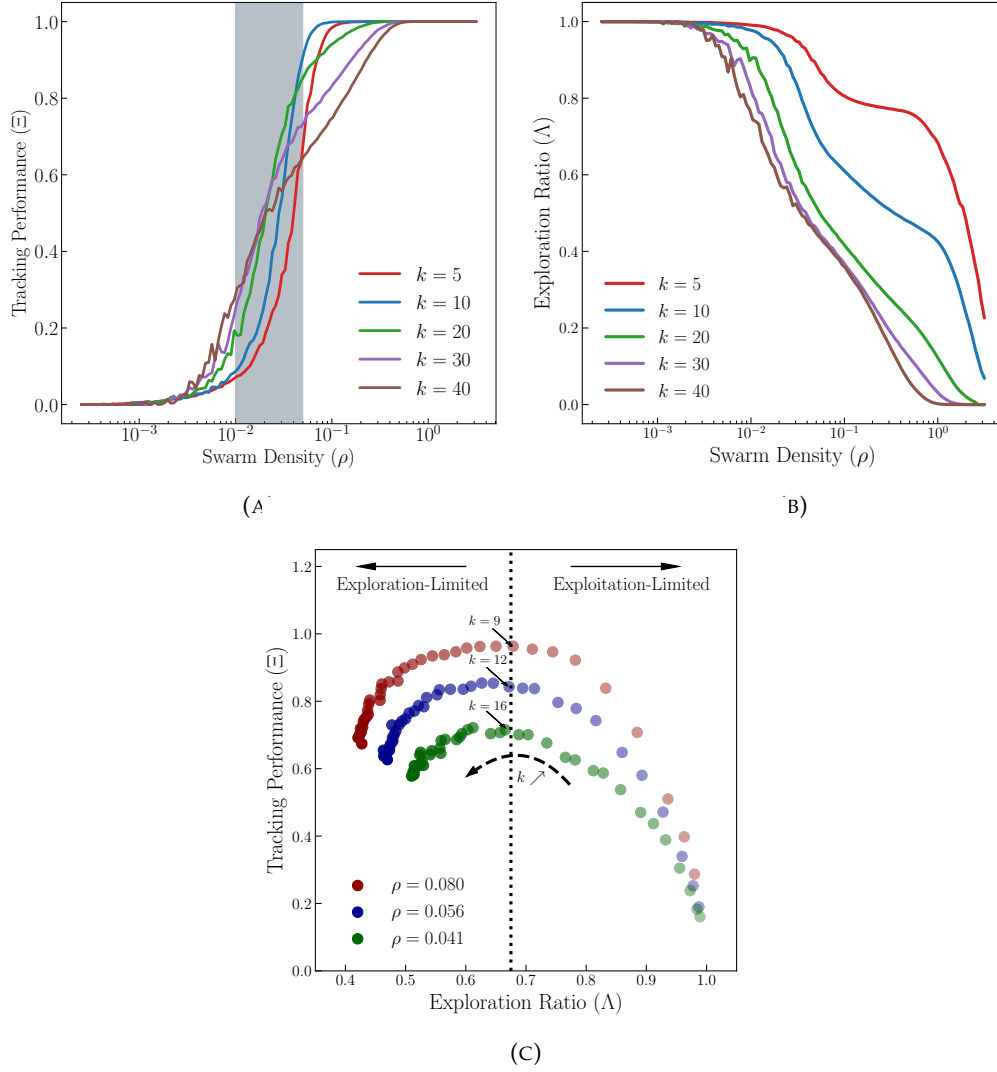


FIGURE 5.4: A swarm of $N = 50$ agents tracking a fast-moving non-evasive target traveling at $v_{o,\max} = 0.15$. 5.4a: Relationship between a swarm's tracking performance (Ξ) and the average swarm density (ρ) while operating at different levels of connectivity (k). The crossover region, where the optimum k maximises Ξ , is highlighted in grey. 5.4b: Relationship between a swarm's exploration ratio (Λ) and the average swarm density while operating at different levels of connectivity. 5.4c: 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, Λ^* , 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 (see Fig. 5.1).

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 able to better exploit this information collaboratively and cluster around the target. This higher level of exploitation allows high k systems to outperform low k systems at low ρ .

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.

The fact that a system’s optimal level of connectivity changes with ρ can be further explained using Fig. 5.4c that shows the presence of an optimum level of exploration Λ^* at which tracking performance is maximised. 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. 5.4a). As such, $\Lambda^* = \Lambda(\max_k(\Xi))$. Although this optimum exploration level, Λ^* , cannot easily be deduced from Fig. 5.4b, it is readily apparent in Fig. 5.4c, indicated by the vertical dotted line. Interestingly, while determining the exact position of Λ^* 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, Λ^* clearly delineates between two regions corresponding to exploration-limited and exploitation-limited systems; 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, which serves as the ‘adaptivity’ lever as discussed earlier, must therefore be adjusted to manage and achieve an optimum balance between exploration and exploitation; a result previously suggested in several studies (Talamali et al., 2021; Horsevad et al., 2022).

5.4.3 Effect of Swarm Size

In previous sections, compelling evidence has been given showing 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, the existence of a complex balance between exploratory and exploitative actions reveals that these emergent properties are strongly dependent on the density. Nonetheless, one should not be put aside unavoidable finite-size effects; for instance, due to the limited number of agents employed within any MRS, usually constrained by financial and logistical challenges (Schroeder, Trease, and Arsie, 2019). As such, the role that a system’s size plays must be acknowledged. It is therefore necessary to complement the presented 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

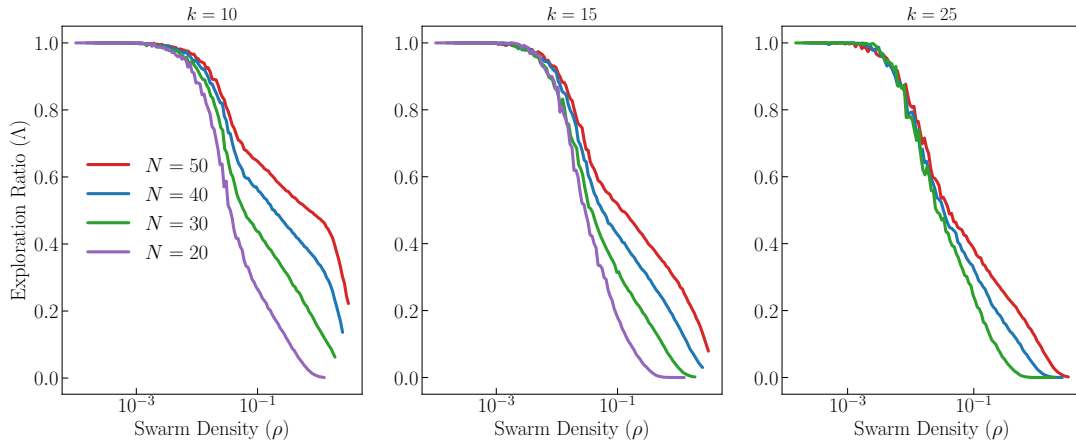


FIGURE 5.5: Exploration ratios of a swarm of $N = 50, 40, 30$, and 20 agents for three different levels of connectivity k and varying swarm densities ρ tracking a fast-moving non-evasive target traveling at $v_{o,\max} = 0.15$.

the level of exploration carried out. Figure 5.5 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. 5.6. 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 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 analysing the tracking performances of differently sized systems. Figure 5.7 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

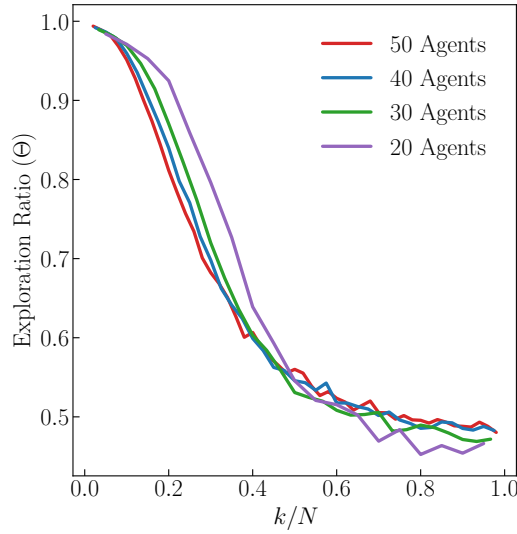


FIGURE 5.6: Exploration ratios of a swarm of $N = 50, 40, 30$, and 20 agents with varying levels of connectivity at a swarm density of $\rho = 0.0444$, tracking a fast-moving non-evasive target traveling at $v_{o,\max} = 0.15$.

constant across systems with different numbers of agents.

Besides the trends observed in exploration and exploitation, Fig. 5.7 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. 5.8 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 localised coordination. However, for the swarms comprised of 30 and 20 agents, tracking performance continues to rise when k is increased and does not display a 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.

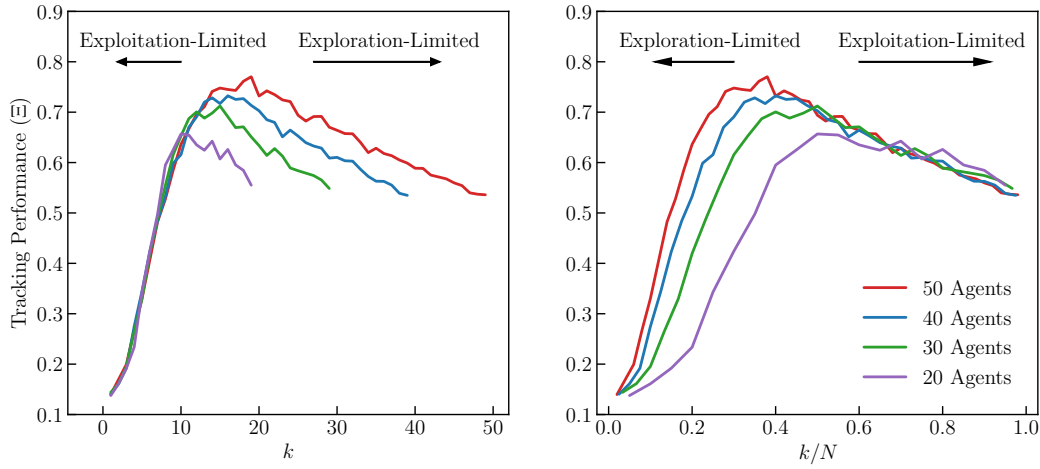


FIGURE 5.7: Tracking performance of a swarm of $N = 50, 40, 30$, and 20 agents with varying levels of connectivity at a swarm density of $\rho = 0.0444$, tracking a fast-moving non-evasive target traveling at $v_{o,\max} = 0.15$. Tracking performances are plot against k (left) and k/N (right).

5.5 Discussion

The work done in this chapter considers a non-evasive tracking task performed over a range of swarm densities. In doing so, the results reveal three density ‘phases’ corresponding to different MRS tracking abilities, namely: (1) a low density phase where the system is simply unable to carry out the tracking task, (2) a high density phase where the system is able to continuously track the target without any interruptions, and (3) an intermediate transition phase between the two where the performance of the system rises very quickly with increasing swarm density. Within this transition phase, it is observed that the value of the optimum level of connectivity, k^* , decreases when the swarm density is increased. This is due to the system’s changing exploration and exploitation dynamics provoked by the density variations. At low density, systems tend to carry out high levels of exploration, leading to systems that have their tracking ability limited through a lack of exploitation, thereby leading to the higher comparative performance of high k systems. Conversely, at high density, systems tend to carry out high levels of exploration and are therefore more likely to be exploration-limited, thereby favouring the use of low k systems.

In addition, the results presented also point to the fact that exploitation is a local task, which depends on the total number of agents carrying out exploitation, while exploration is a global one, which depends on the proportion of agents exploring the environment. As such, when comparing a large swarm to a small one operating at the swarm density, the large swarm will be expected to out perform the smaller system due to its ability to allocate more agents to carry out exploitation of a target’s positional information, yet still retain a high proportion of agents exploring the environment.

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

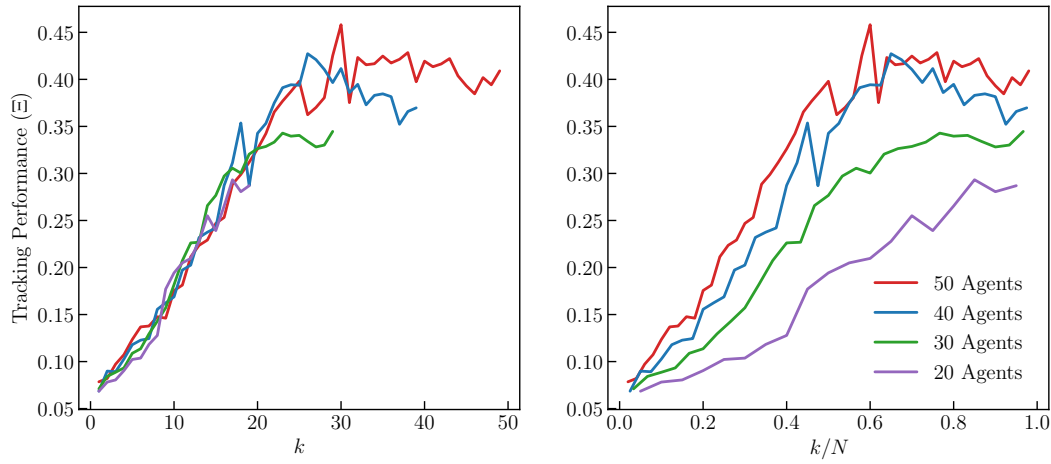


FIGURE 5.8: Tracking performance of a swarm of $N = 50, 40, 30$, and 20 agents with varying levels of connectivity at a swarm density of $\rho = 0.02$, tracking a fast-moving non-evasive target traveling at $\mathbf{v}_{o,\max} = 0.15$. Tracking performances are plot against k (left) and k/N (right).

robots, the MRS inevitably fails to coordinate the actions of individual units, thereby hindering the emergence of a robust collective response. On the other hand, with too many robots, unavoidable interference in the coordination of behaviours yields a sub-optimal performance and a waste of resources. 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, or even non-existent, collective action to a maximum in collective performance. For the collective task of target tracking, the work presented in this chapter has clearly found such an interval: the aforementioned transition phase. From the perspective of an MRS practitioner, 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, thereby taking a fuller advantage of the system's swarm intelligence and at the same time, reducing the cost of the system.

Chapter 6

Conclusions and Future Outlook

6.1 Summary of Findings

The work presented in this thesis was done to answer the the main research question, “How can an decentralised MRS be made to find and track a target that can move faster than any of its component robots?” To this end, a novel strategy that can be used by a decentralized MRS in a target search and track task was developed. This strategy is unlike any approaches currently used in the literature as it allows the system to track targets that can move faster than any of the system’s component units. With the proposed method, an MRS is also able to cope with the simultaneous tracking of multiple targets, as well as targets that move in both evasive and non-evasive fashions. The strategy is comprised of two components: (1) a component that attracts agents towards a target’s position, which was inspired by the Social-Only PSO algorithm first proposed by Engelbrecht (2010), and (2) an adaptive repulsion component that promotes area exploration, as well as serves as an anti-collision measure. The use of this strategy is facilitated through the use of a topological k -nearest neighbour communications network that allows a user to change an agent’s number of communications neighbours. In the pursuit of an evasive target, the attractive component is augmented with an agent-based short-term memory, thereby allowing agents to aggregate around a location based on a target’s position history. While most of the work done in this thesis was performed in virtual simulations, the key behaviours crucial to an MRS’s tracking performance, namely the aggregation around a target’s position and inter-agent repulsion, and their effects on the system performance were also demonstrated through their implementation in a physical MRS. The results presented in this thesis show that the search and tracking of fast-moving targets need not be limited to the domain of MRS with infinite sensor range, as was the case for Janosov et al. (2017) and that the presented strategy facilitates the tracking of such targets in a wide range of scenarios. Indeed, the newly proposed strategy was shown to be able to outperform other benchmark strategies commonly found in the current literature.

Parts of the content in this chapter is intended for publication in:
Kwa, H.L., Philippot, J., and Bouffanais, R. [In Review]. “Effect of Swarm Density on Collective Tracking Performance”. In: *Swarm Intelligence*.

When studying the effects of varying a system's parameters on its tracking performance and exploration-exploitation dynamics (EED), it was established that changing an agent's level of connectivity, k , was the primary means that allows one to change the swarm's EED. In increasing k , the system more readily disseminates a target's positional information among its component agents, thereby making the system more inclined to cluster around a target and carry out exploitation. In contrast, reducing k also limits the flow of information around the agents. Due to this lack of information, the adaptive repulsion component of the proposed strategy forces the individual robots to explore the environment in search of a target, increasing the system's level of exploration. In addition, the introduction of agent-based memory also served to increase the system's level of exploitation. However, it was also shown that increasing the length of an agent's memory past a certain level had a smaller impact on the system's EED compared to when the system's k was varied. In addition, excessive memory lengths also resulted in poorer tracking performances due to the swarm exploiting outdated information.

The key to maximising a system's tracking performance is to find the correct EED balance. To optimise its performance, the MRS must carry out adequate levels of exploitation, allowing the system to follow the target, and exploration, allowing the system to reacquire the target's position after being outrun. As mentioned previously, this is primarily achieved by changing k , although it can also be done to a limited extent by changing an agent's memory length. The position of this balance is heavily dependent on the task assigned to the system. More difficult tasks, such as those involving tracking evasive targets, as well as those involving the simultaneous tracking of multiple targets required, required higher levels of exploration. This is because such targets tend to outrun the system more often, thereby requiring the MRS to carry out more exploration to reacquire the targets' locations. Hence, maximum performance for these tasks require the system to use lower levels of k . In contrast, less challenging tasks involving relatively slower, single, and non-evasive targets allow agents to follow the targets for longer. In such cases, the system should be setup to favour the conduct of exploitative activities and using low k values would promote excessive levels of exploration. These results go against the commonly held belief that increasing a system's level of connectivity leads to improved levels of performances. The work in this thesis has shown that to maximise a system's performance, its k and an agent's memory length needs to be assigned to solicit the required amount of exploration and exploitation for a given task.

Besides the target parameters, the swarm density also has a large role to play in the system's EED. In varying the swarm density, three distinct phases were demonstrated: (1) a low density phase where the system is simply unable 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. The most important of these three phases is the transition phase where it is clear that increasing the system density also increases the MRS's propensity to carry out exploitation. This is because within this phase, reducing inter-agent separation makes it easy for units to cluster together,

thereby facilitating exploitation. The inverse is also true; increasing inter-agent separation discourages clustering, thereby facilitating exploration. In the results presented, it is evident that at low densities, systems tend to be exploitation-limited while systems operating at high densities tend to be exploration-limited. The same results also demonstrated that the optimum exploration and exploitation balance appears to remain the same while carrying out similar tasks regardless of the density. This suggests that the strategy used by an MRS must be set up in a manner that takes into account the level of exploration and exploitation encouraged by the swarm density in addition to the requirements imposed by the set task.

In addition to the swarm density, one must also account for the size of the MRS that determines its capacity to carry out exploratory and exploitative actions. When varying the size of a system at a constant density, it was shown that exploitation is a localised activity, which dependant on the total number of agents carrying out the activity. This is in contrast to exploration, which is a system-wide activity dependant on the proportion of agents exploring the environment. Therefore, a large system with more agents has the capacity to carry out high amounts of exploration and maintain a high level of exploitation. This is opposed to smaller systems that see larger decreases in their level of exploratory behaviour when committing agents to the conduct of exploitation. Thus, it can be said that when increasing k , small systems tend to become exploration limited much quicker and have lower maximum levels of performance than their larger counterparts even when operating at the same swarm density.

Finally, the use of heterogeneous systems, which include faster moving agents, together with the newly introduced strategy was also studied. In doing so, it was shown that unlike several other works, implementing the search and track strategy in a heterogeneous swarm is able to yield improved tracking performances without the need for modifications. This occurs because the faster moving units are able to engage with the target, thereby better exploiting the target's positional information, and track it for longer periods of time before being outrun. The higher mobility and ability to engage with the target of the fast agents increases the system's propensity to carry out exploitative activity shifts the optimal EED for the heterogeneous system more in favour of exploitation. However, to obtain this optimal EED balance, the increased exploitation tendencies of the heterogeneous swarm needs to be tempered by reducing the level of communications between agents, leading to a lower k^* .

6.2 Discussion and Outlook

The work done in this thesis focused mainly on using a decentralised strategy to perform target tracking task. By using this strategy, a practitioner can vary the exploratory and exploitative behaviours of the system by manually selecting a degree, k , of the topological k -nearest neighbour network, or by varying the length of the agent-based memory. This thesis also looked at the effect of swarm density on the system's EED balance. While important, this is only a small contribution to the MRS community and there are many other areas that need to be studied to allow for the more widespread adoption of swarming MRS.

Firstly, more studies need to be carried out on communications in swarm robotic systems. Throughout this thesis, a topological k -nearest neighbour network was used. This choice was explicitly made as the response of a swarm is influenced by the number of communication neighbours an agent has, rather than the number of neighbours within a given radius (Mateo, Kuan, and Bouffanais, 2017; Horsevad et al., 2022). Using this type of network permitted the study of the effects on the system's performance and EED when each agent added one more neighbour. While doing so has revealed an optimal level of connectivity for a given task, care must be taken when implementing this type of network within a physical swarm. The simulations were run on the assumption of perfect communications; agents were able to communicate with each other without information loss or dropped connections, regardless of the distance between the two. This assumption may not always be true, especially when a mission takes place in a large environment or when there are large amounts of noise affecting radio communications. In such cases, the communication ranges of the individual robot units and the quality of inter-robot radio transmissions must be considered. Should the number of neighbours within communications range fall under the optimal level of connectivity or information be lost during information transmission, measures need to be developed that increase the amount of exploitation carried out by the swarm.

Along the same vein, in this work, the used MRS are initialised with a k value that is maintained throughout the simulation or experiment; i.e., while the agents are able to change which neighbours they communicated with, the number of neighbours is always kept constant. While this may work well in scenarios where the system designer has *a priori* knowledge of the environment, e.g., the size of the environment, as well as the speed and movement policy of the target(s) are known before hand, the system will not perform at its optimum should it operate in a dynamic environment. This includes situations where the size of the area of operations grows and shrinks over time (Vallegra et al., 2018), as well as when the objects or number of objects to be tracked changes over the course of the mission (Esterle, 2018a; Esterle and King, 2022). To improve the performance of an MRS in a scenario with evolving task parameters, an agent could be allowed to modulate its individual value of k . This can be done by considering local factors, such as the amount of time that has elapsed since an agent has tried engaging with a target, as well as the state of an agent's neighbours. Giving each agent the ability to modulate its own k gives the MRS a higher degree of decentralisation and adaptivity, hence making it more flexible in the face of changing mission parameters as the individual units are more readily able to adapt to its local conditions.

Alternatively, different information dissemination methods, such as the use of simple or complex contagion (Horsevad, 2021), can be explored. However, should any form of contagion be used, new methods need to be developed that allow an agent to determine when and how to engage with a target; an agent cannot simply attempt to track a target once it has information of the target's location, as is done in the currently proposed strategy. This is because the use of contagion causes information to spread very quickly across the system, especially in cases where the size of the system is comprised of less than 100 units (Balister et al., 2005; Komareji and Bouffanais, 2013). As a result of this rapid spread of information, more 'intelligent' movement patterns and information processing algorithms may need to be developed that prevents excessive exploitative

behavior.

Another assumption made was the absence of obstacles blocking the path of agents; both simulations and physical experiments were carried out within an environment devoid of obstacles. In a more realistic scenario, the presence of obstacles will result in the hindrance of the tracking process. While work has been done studying the use of target tracking algorithms in environments with sparsely distributed obstacles (Dadgar, Jafari, and Hamzeh, 2016), these works only consider targets that move slower than the pursuing agents. As such, more work needs to be performed studying the tracking of fast-moving targets in the presence of obstacles before their implementation into an operational system. Along the same vein, different types of environments should also be studied, such as the operation of the swarm in obstacle-dense environments (Li et al., 2020) or in the presence of mobile obstacles (Sun et al., 2022).

While one of the key characteristics of swarm robotics is the robustness of the system, the work done in this thesis did not study the cases where individual robot units suddenly fail or when a robot's sensors are affected by the presence of noise. At best, these factors would reduce the system's overall level of exploitation when a target is encountered, and at worst, the system may end up aggregating in the wrong location due to incorrect information stemming from one or a group of robots, thereby deteriorating the MRS's performance. Investigating the behaviour and performance of an MRS using imperfect sensors and in the face of individual unit failures will allow for more resilient strategies to be developed, making the use of such systems more feasible in real world conditions.

When dealing with heterogeneous systems, the work done in this thesis only considered the case where more mobile agents with higher maximum speeds were substituted into the swarm. However, system heterogeneity does not only stem from an agent's movement abilities. The aquatic surface drone system developed by Christensen et al. (2015) utilises two sets of drones with different communication ranges. The subCUL-Tron swarm designed by Thenius et al. (2016) is comprised of agents with different complementing functionality. Further study into heterogeneity could lead to the development of heterogeneous behaviour, allowing a practitioner to take advantage of the differences between two of more agent types.

Over the course of this thesis, the importance of attaining the correct exploration-exploitation balance to optimise a system's performance has been emphasised. However, as evidenced by the various results presented, this optimum balance changes according to the assigned task's parameters. Even if the optimum balance stays constant, variations in an MRS's environment can push the system to over-exploit information or over-explore the environment. At this point, it is worth remembering that one of the key tenets of swarm robotics is a system's flexibility. As defined by Dorigo, Theraulaz, and Trianni (2021), this is "the capacity to solve problems/perform tasks that depart from those chosen at design time." The absence of an absolute metric to measure a system's flexibility has necessitated the use 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. In the work done in this thesis, a system's flexibility was tested in multiple ways, such as by changing the targets' speed and movement profile, as well as the number of targets to be tracked.

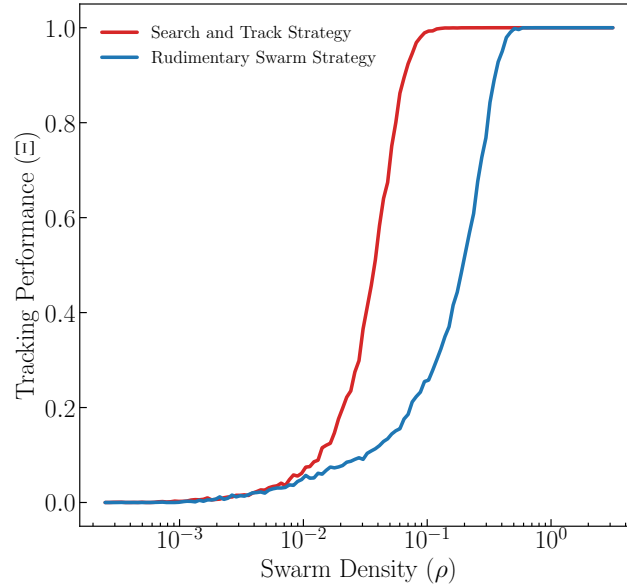


FIGURE 6.1: Comparison the search and track strategy presented in Chapter 4 and a rudimentary memory-less strategy that uses a constant inter-agent repulsion behaviour. Both sets of simulations were carried out at a level of connectivity, $k = 10$. The performance curve for the rudimentary strategy is shifted towards a higher density range compared to the baseline swarm strategy.

The strategies presented in this thesis afforded MRS a certain level of flexibility; the systems were still able to perform target tracking to a certain degree when the task conditions (e.g., density, target speed, etc.) were changed. However, the only way to maximise a system's collective tracking performance across a wide range of situations is to allow it to autonomously modify its exploration-exploitation balance by tapping into adaptivity. As defined by Dorigo, Theraulaz, and Trianni (2021), this is the MRS's "ability to learn/change behaviour to respond to new operating conditions." The power of a system's adaptivity is clearly illustrated in Fig. 6.1, where the proposed search-and-track swarm strategy with $k = 10$ (Chapter 4) is shown alongside a strategy, which is identical in all ways except that it is memory-less and has a fixed repulsion strength, i.e., a strategy that facilitates a lower degree of system adaptivity. 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 taps into adaptivity that yields a more swarm-intelligent collective tracking of the target. Hence, in this context it appears that adaptivity promotes

flexibility.

In recent years, the field of multi-agent reinforcement learning (MARL) has become increasingly popular, including their application within the context of swarm robotics (Hüttenrauch, Šošić, and Neumann, 2018; Hüttenrauch, Oic, and Neumann, 2019; Kouzehgar, Meghjani, and Bouffanais, 2020). Buoyed by successes of MARL evolving task-specific robot controllers (Gomes, Urbano, and Christensen, 2013; Duarte et al., 2016), practitioners are starting to look towards automatic methods in the design of robot swarms (Francesca and Birattari, 2016; Bozhinoski and Birattari, 2021). Currently, such methods have cast their scenarios and missions into optimisation problems. While this may yield very good performances when all relevant task parameters are known in advance, the work presented in this thesis serves as a cautionary tale; system performances may experience significant degradation if there are sudden changes to the operating environment or mission parameters due to the variations in the level of exploration and exploitation demanded. Instead of designing swarms to accomplish very specific tasks in equally specific environments, research on MARL in swarms should instead be directed towards optimising systems for adaptivity over a wide range of conditions. The work done in this thesis has identified some of the parameters that can be optimised to make swarms more adaptive, namely, a swarm's degree of connectivity and the length of an agent's memory. Allowing MRS to vary such parameters will allow them to change their behaviours and learn new ones according to the time-specific demands of the task at hand, thereby opening up promising new opportunities and avenues of research for the community working on the design of effective collective swarm dynamics.

Appendix A

Local Density

This section shows the reasoning why Equation 5.3 was used in the calculation of the system's local swarm density. This is restated here for completeness:

$$\rho_L = \frac{1}{NT} \sum_{t=1}^T \sum_{i=1}^N \frac{7}{\pi L_i^2} \quad (\text{A.1})$$

Due to the implemented inter-agent repulsion behavior, agents will tend to fall into a hexagonal packing pattern as seen in Fig. A.1a. 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. A.1b. 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 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.

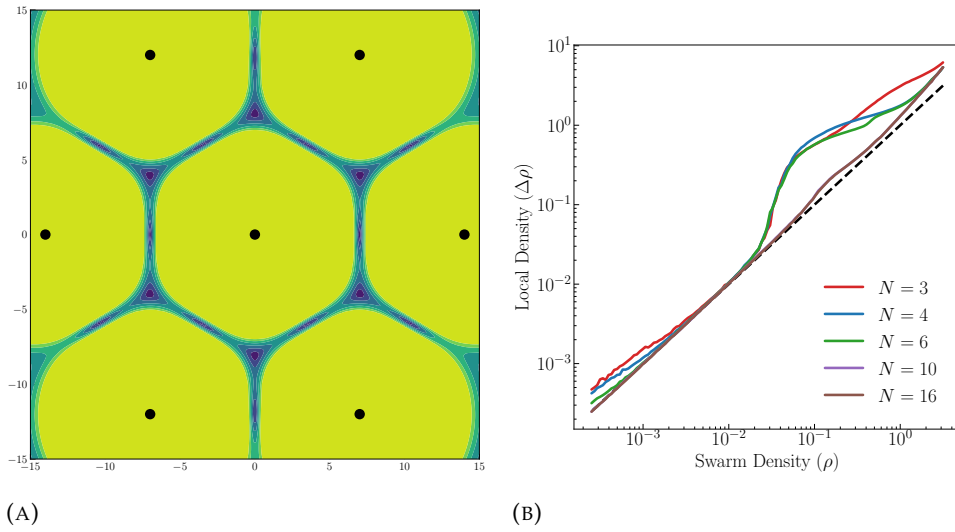


FIGURE A.1: (A) Positioning of agents (black dots) in relation to each other and the repulsion fields generated by the individual agents while in their equilibrium positions. Areas in yellow represent areas of high repulsion potential while those in blue represent areas of low repulsion potential. Given 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).

Appendix B

Factors Affecting the Transition Phase

As seen in Chapter 5, changing a swarm's level of connectivity changes the range densities over which the transition phase occurs, characterised by a sharp rise in the system's tracking performance. The work presented in this appendix explores how changing the tracking task and system parameters affects the range of transition densities.

From Fig. B.1a, it is evident that both the start and end of the transition phase occur at a lower swarm density when tracking a non-evasive target compared to when tracking an evasive target. This is to be expected as tracking a non-evasive target is inherently an easier task than tracking an evasive target as the target is not actively attempting to avoid its pursuing agents. As such, when the target is found, the swarming agents are able to follow it for longer, allowing agents to continue broadcasting the target's positional information for longer. With target information being more available to the system, agents are able to stay engaged with the target for longer and carry out effective exploitation of the target's positional information, thereby lowering the transition density.

Interestingly, when comparing the level of exploration and exploitation carried out in the two cases, both evasive and non-evasive tracking MRS demonstrate a similar level of EED balance when operating in the low and transition density phases, with the exploitation ratio curves only diverging after reaching the high density phase. This suggests that in the two cases, the system carries out the same level of exploration and exploitation. However, the evasiveness of the target plays a key role in determining the effectiveness of the system's exploitative activity.

When varying the maximum speed of the target, Fig. B.1c illustrates that lowering the target's maximum speed has the effect of also lowering the density at which the transition phase starts. Like the comparisons between evasive and non-evasive target tracking, tracking a slower target is an easier task. As such, an individual agent is able to follow the target for longer, thereby allowing it to transmit target information to its neighbors for longer. This in turn allows the system to increase its engagement ratio, Θ , at lower swarm densities, resulting in a lower transition density.

When tracking multiple targets, as seen in Fig. B.1e, the transition phase starts at a lower density compared to the single target tracking case. This is especially true in systems that use a high level of inter-agent connectivity for reasons explored in Chapter 5.

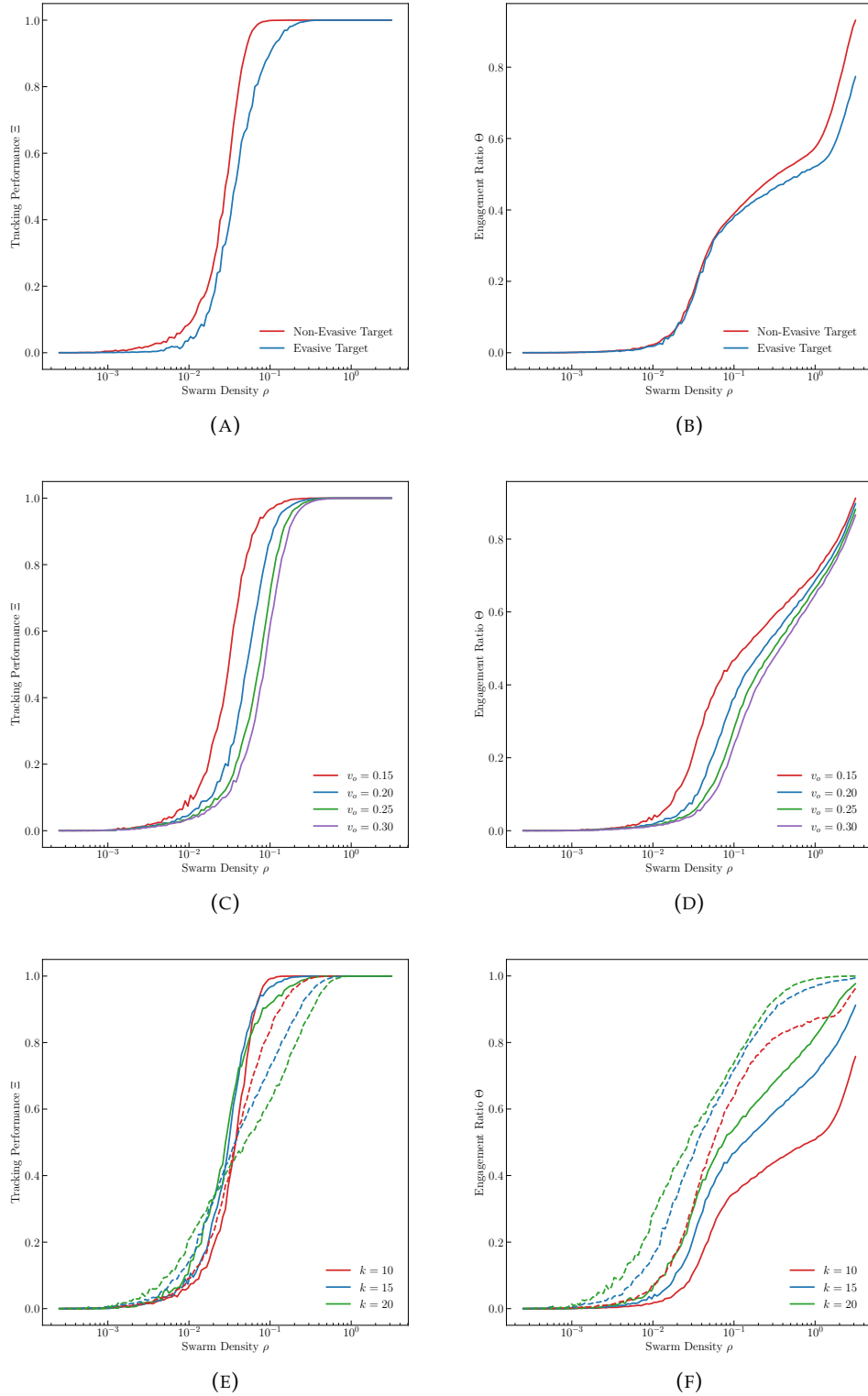


FIGURE B.1: The tracking performance and engagement ratio of swarm of 50 agents tracking different types of targets. (A) and (B): Tracking a single evasive and non-evasive target with $\mathbf{v}_{o,\max} = 0.15$ and level of connectivity set at $k = 15$. (C) and (D): Tracking targets with varying $\mathbf{v}_{o,\max}$ that follow a non-evasive movement policy. with the level of connectivity set at $k = 15$. (E) and (F): Single target tracking (solid line), triple target tracking (dotted line) with various levels of connectivity. The target travels at a maximum speed of $\mathbf{v}_{o,\max} = 0.15$ following a non-evasive movement policy.

The presence of multiple targets means that an agent is more likely to encounter one of the targets as it explores the search space. As such, the combined presence of the three targets will provoke a much higher level of exploitative activity as compared to a single target, as seen Fig. B.1f. This results in the swarm being able to track at least one of the three targets at lower densities and an increased tracking performance.

This happens as the strategy employed by the agents does not differentiate between targets; an agent will make its movements based on the most up-to-date information it receives. Therefore, a system with a high level of connectivity will tend to have a large proportion of agents exploiting the same piece of information, thereby forming a large cluster around a single target. This means that system will be able to track one target very effectively, thereby contributing to the higher system performance at low swarm density. However, this advantage turns in favour of systems with low levels of connectivity at high swarm densities. In contrast to systems operating at high k , systems that operate at low k are able to form multiple clusters that exploit the multiple information sources and surround the different targets. This allows multiple targets to be tracked simultaneously, meaning that the performance of systems with low k is better at high swarm densities.

Appendix C

Result Variability

As mentioned in the main text, to obtain the data points to generate the result plots, each simulation run was carried out over the course of 100,000 iterations. This led to a very low level of variability in the obtained results of $< 1\%$ when different random seeds were used. Due to this low level of variability obtained from the simulation results, the uncertainty ranges were not included in the figures shown in the main text. A sample of figures with the uncertainty range have been included in this section of the appendix. In the figures that follow, the range of uncertainty is illustrated by the shaded area above and below the main line.

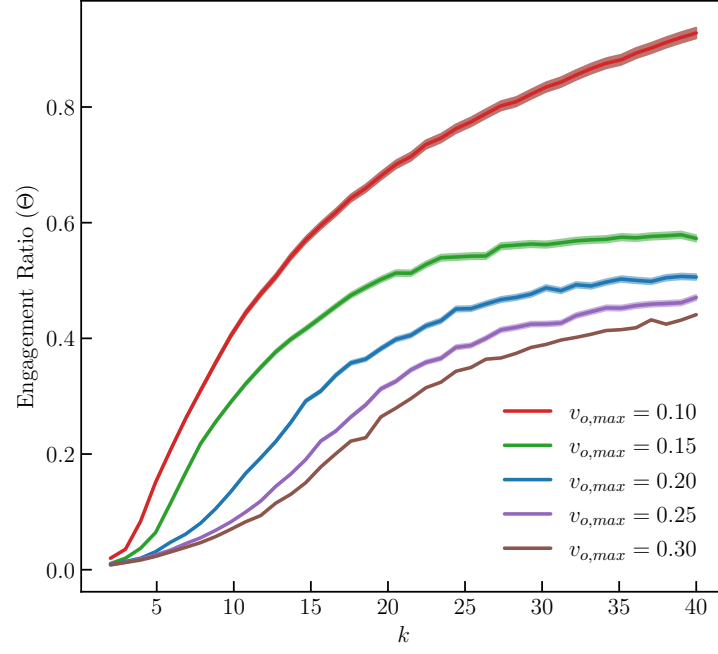


FIGURE C.1: Companion figure to Fig. 3.5 with uncertainty ranges included.

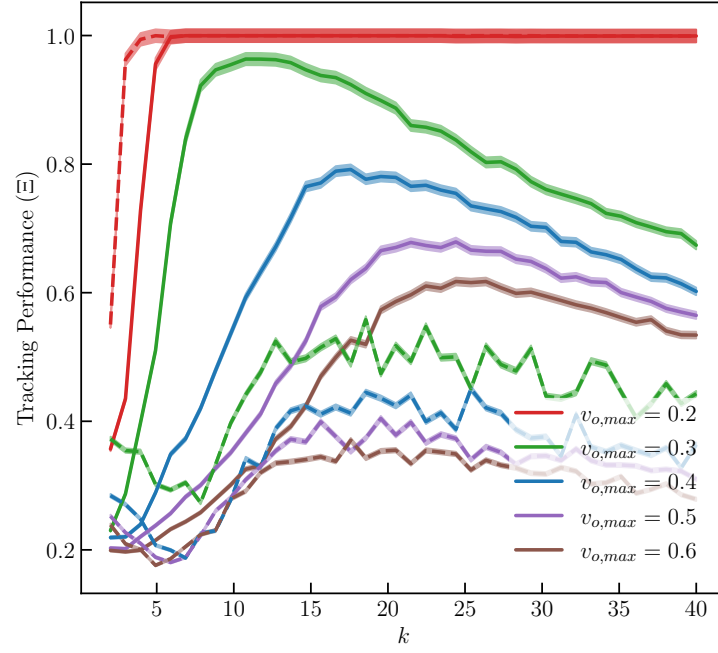


FIGURE C.2: Companion figure to Fig. 3.7 with uncertainty ranges included.

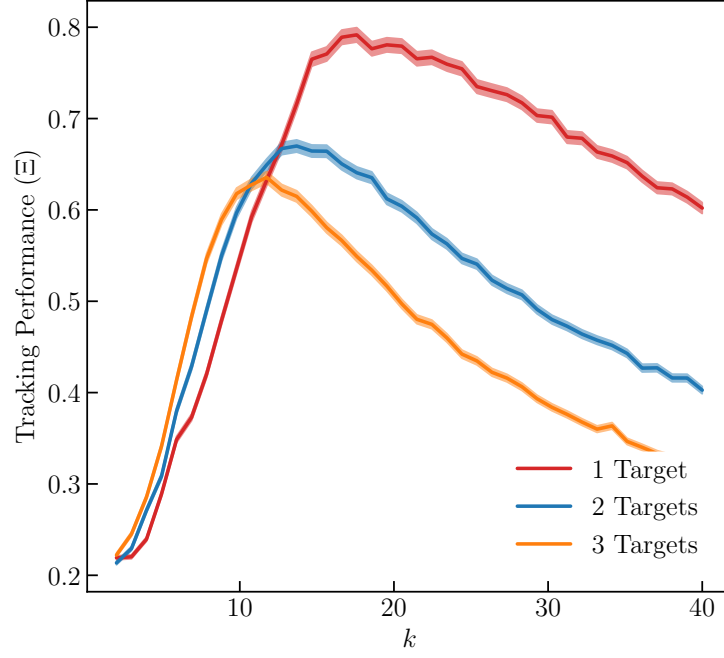


FIGURE C.3: Companion figure to Fig. 3.8 with uncertainty ranges included.

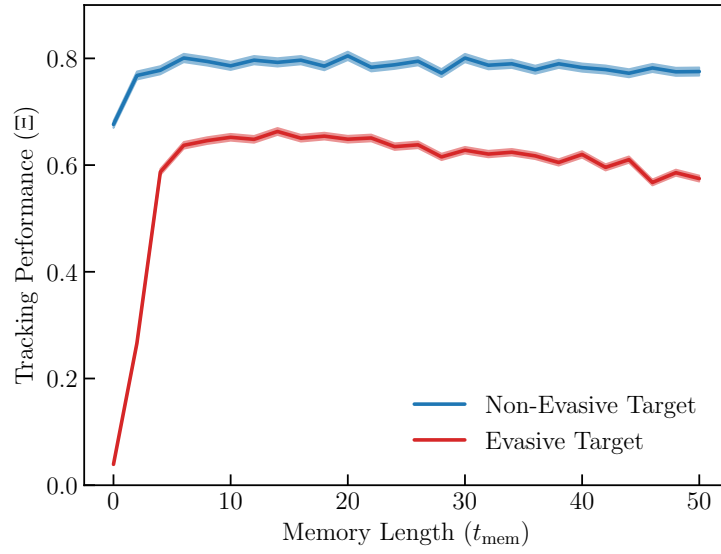


FIGURE C.4: Companion figure to Fig. 4.5 with uncertainty ranges included.

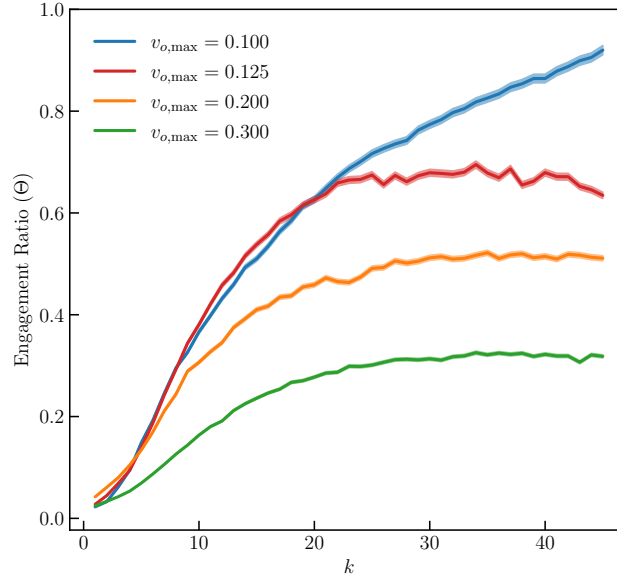


FIGURE C.5: Companion figure to Fig. 4.3 with uncertainty ranges included.

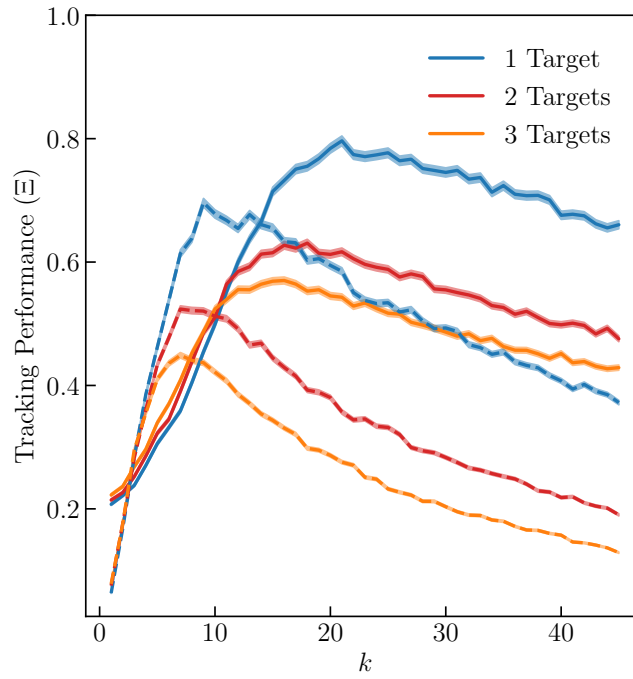


FIGURE C.6: Companion figure to Fig. 4.4 with uncertainty ranges included.

Bibliography

- Açikmeşe, Behçet and David S Bayard (2012). "A Markov Chain Approach to Probabilistic Swarm Guidance". In: *2012 American Control Conference (ACC)*. Montreal, Quebec, Canada: IEEE, pp. 6300–6307. ISBN: 9781457710964. DOI: 10.1109/ACC.2012.6314729.
- Althshuler, Yaniv (2009). "To Add With Caution — Decreasing a Swarm Robotics' Efficiency by Imprudently Enhancing the Robots' Capabilities". In: *ICARA 2009 - Proceedings of the 4th International Conference on Autonomous Robots and Agents*. Wellington, New Zealand: IEEE, pp. 351–356. ISBN: 9781424427130. DOI: 10.1109/ICARA.2000.4804014.
- Alvarenga, Carlos Diaz, Nicola Basilico, and Stefano Carpin (2020). "Multirobot Patrolling Against Adaptive Opponents with Limited Information". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. Paris, France, pp. 2486–2492. ISBN: 9781728173955.
- Attanasi, Alessandro et al. (2014). "Collective Behaviour without Collective Order in Wild Swarms of Midges". In: *PLoS Computational Biology* 10.7, pp. 1–10. ISSN: 15537358. DOI: 10.1371/journal.pcbi.1003697.
- Azad, S Kazemzadeh and OĞUZHAN Hasançebi (2014). "An elitist self-adaptive step-size search for structural design optimization". In: *Applied Soft Computing* 19, pp. 226–235. DOI: 10.1016/j.asoc.2014.02.017.
- Azoulay-Schwartz, Rina, Sarit Kraus, and Jonathan Wilkenfeld (2004). "Exploitation vs. exploration: Choosing a supplier in an environment of incomplete information". In: *Decision Support Systems* 38.1, pp. 1–18. ISSN: 01679236. DOI: 10.1016/S0167-9236(03)00061-7.
- Babazadeh, Abbas, Hossain Poorzahedy, and Saeid Nikoosokhan (2011). "Application of particle swarm optimization to transportation network design problem". In: *Journal of King Saud University-Science* 23.3, pp. 293–300. DOI: 10.1016/j.jksus.2011.03.001.
- Balister, Paul et al. (2005). "Connectivity of random k-nearest-neighbour graphs". In: *Advances in Applied Probability* 37.1, pp. 1–24. DOI: 10.1239/aap/1113402397.
- Ballerini, Michele et al. (2008). "Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study". In: *Proceedings of the National Academy of Sciences* 105.4, pp. 1232–1237. ISSN: 0027-8424. DOI: 10.1073/pnas.0711437105. eprint: 0709.1916.
- Banfi, Jacopo et al. (2015). "Fair Multi-Target Tracking in Cooperative Multi-Robot systems". In: *Proceedings - IEEE International Conference on Robotics and Automation*. Seattle, Washington, USA, pp. 5411–5418. DOI: 10.1109/ICRA.2015.7139955.
- Bayindir, Levent (2016). "A review of swarm robotics tasks". In: *Neurocomputing* 172, pp. 292–321. ISSN: 18728286. DOI: 10.1016/j.neucom.2015.05.116.

- Beni, Gerardo and Jing Wang (1993). "Swarm Intelligence in Cellular Robotic Systems". In: *Robots and Biological Systems: Towards a New Bionics? 2*, pp. 703–712. DOI: 10.1007/978-3-642-58069-7_38.
- Bennett, Chris, Seth Bullock, and Jonathan Lawry (2021). "Demonstrating the Differential Impact of Flock Heterogeneity on Multi-agent Herding". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 13054 LNAI, pp. 147–157. ISSN: 16113349. DOI: 10.1007/978-3-030-89177-0_15.
- Berger-Tal, Oded et al. (2014). "The exploration-exploitation dilemma: A multidisciplinary framework". In: *PLoS ONE* 9.4. ISSN: 19326203. DOI: 10.1371/journal.pone.0095693.
- Biswal, Shiba, Karthik Elamvazhuthi, and Spring Berman (2021). "Decentralized Control of Multi-Agent Systems using Local Density Feedback". In: *IEEE Transactions on Automatic Control*. DOI: 10.1109/TAC.2021.3109520.
- Blackwell, Tim (2007). "Particle swarm optimization in dynamic environments". In: *Evolutionary computation in dynamic and uncertain environments*. Springer, pp. 29–49.
- Blackwell, Tim and Peter Bentley (2002). "Dynamic Search with Charged Swarms". In: *Proceedings of the Genetic and Evolutionary Computation Conference*. New York City, NY, USA, pp. 19–26. ISBN: 1-55860-878-8. DOI: 10.1.1.97.9635.
- Blackwell, Tim and Jürgen Branke (2006). "Multiswarms, Exclusion, and Anti-Convergence in Dynamic Environments". In: *IEEE Transactions on Evolutionary Computation* 10.4, pp. 459–472. DOI: 10.1109/TEVC.2005.857074.
- Blackwell, Tim and Jim Kennedy (2018). "Impact of communication topology in particle swarm optimization". In: *IEEE Transactions on Evolutionary Computation* 23.4, pp. 689–702. ISSN: 1089778X. DOI: 10.1109/TEVC.2018.2880894.
- Bodi, Michael et al. (2012). "Interaction of robot swarms using the honeybee-inspired control algorithm BEECLUST". In: *Mathematical and Computer Modelling of Dynamical Systems* 18.1, pp. 87–100. ISSN: 13873954. DOI: 10.1080/13873954.2011.601420.
- Bodi, Michael et al. (2015). "BEECLUST used for exploration tasks in autonomous underwater vehicles". In: *IFAC-PapersOnLine* 48.1, pp. 819–824. DOI: 10.1016/j.ifacol.2015.05.133.
- Bonyadi, Mohammad Reza and Zbigniew Michalewicz (2017). "Particle Swarm Optimization for Single Objective Continuous Space Problems: A Review". In: *Evolutionary Computation* 25.1, pp. 1–54. DOI: 10.1162/EVCO_r_00180.
- Bouffanais, Roland (2016). *Design and Control of Swarm Dynamics*. Singapore: Springer Singapore. ISBN: 978-981-287-750-5. DOI: 10.1007/978-981-287-751-2.
- Bozhinoski, Darko and Mauro Birattari (2021). "Towards an integrated automatic design process for robot swarms". In: *Open Research Europe* 1, p. 112. DOI: 10.12688/openreseurope.14025.1.
- Brambilla, M et al. (2013). "Swarm robotics : A review from the swarm engineering perspective". In: *Swarm Intelligence* 7.1, pp. 1–41. ISSN: 1935-3812. DOI: 10.1007/s11721-012-0075-2.
- Cao, Xiang and Changyin Sun (2017). "A potential field-based PSO approach to multi-robot cooperation for target search and hunting". In: *At-Automatisierungstechnik* 65.12, pp. 878–887. ISSN: 2196677X. DOI: 10.1515/auto-2017-0080.

- Cao, Xiang and Changyin Sun (2018). "Multi-AUV cooperative target hunting based on improved potential field in underwater environment". In: *2018 33rd Youth Academic Annual Conference of Chinese Association of Automation (YAC)*. IEEE, pp. 118–122. DOI: 10.1109/YAC.2018.8406357.
- Carlisle, Anthony and Gerry Dozier (2000). "Adapting particle swarm optimization to dynamic environments". In: *International Conference on Artificial Intelligence*, pp. 429–434.
- Cates, Michael E. and Julien Tailleur (2015). "Motility-induced phase separation". In: *Annual Review of Condensed Matter Physics* 6.1, pp. 219–244. ISSN: 19475462. DOI: 10.1146/annurev-conmatphys-031214-014710. eprint: 1406.3533.
- Cavagna, A. et al. (2010). "Scale-free correlations in starling flocks". In: *Proceedings of the National Academy of Sciences* 107.26, pp. 11865–11870. ISSN: 0027-8424. DOI: 10.1073/pnas.1005766107.
- Chen, Xin-Xing and Jian Huang (2019). "Odor source localization algorithms on mobile robots: A review and future outlook". In: *Robotics and Autonomous Systems* 112, pp. 123–136. ISSN: 09218890. DOI: 10.1016/j.robot.2018.11.014.
- Chen, Zhifu, Huimin Liao, and Tianguang Chu (2011). "Clustering in multi-agent swarms via medium-range interaction". In: *EPL (Europhysics Letters)* 96.4, p. 40015. DOI: 10.1209/0295-5075/96/40015.
- Cheng, Shi et al. (2015). "Population Diversity Maintenance In Brain Storm Optimization Algorithm". In: *Journal of Artificial Intelligence and Soft Computing Research* 4.2, pp. 83–97. ISSN: 2083-2567. DOI: 10.1515/jaiscr-2015-0001.
- Christensen, Anders Lyhne et al. (2015). "Design of Communication and Control for Swarms of Aquatic Surface Drones". In: *ICAART 2015 - Proceedings of the International Conference on Agents and Artificial Intelligence*. Lisbon, Portugal, pp. 548–555. ISBN: 978-989-758-073-4. DOI: 10.5220/0005281705480555.
- Coogler, Richard A. and Ayanna M. Howard (2013). "The iceberg observation problem: Using multiple agents to monitor and observe ablating target sources". In: *Proceedings - 2013 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2013*. Manchester, United Kingdom: IEEE, pp. 1660–1665. ISBN: 9780769551548. DOI: 10.1109/SMC.2013.286.
- Coquet, Charles, Andreas Arnold, and Pierre-Jean Bouvet (2021). "Control of a robotic swarm formation to track a dynamic target with communication constraints : analysis and simulation". In: *Applied Sciences* 11.3179. DOI: 10.3390/app11073179.
- Coquet, Charles et al. (2019). "A Local Charged Particle Swarm Optimization to Track an Underwater Mobile Source". In: *OCEANS 2019 - Marseille*. Marseille, France: IEEE. DOI: 10.1109/OCEANSE.2019.8867527.
- Couceiro, Micael S and David Portugal (2018). "Swarming in forestry environments: collective exploration and network deployment". In: *Swarm Intell. Princ. Curr. Algorithms. Methods* 119, p. 323. DOI: 10.1049/PBCE119F_ch12.
- Couceiro, Micael S, Rui P Rocha, and Nuno MF Ferreira (2011a). "A novel multi-robot exploration approach based on particle swarm optimization algorithms". In: *2011 IEEE International Symposium on Safety, Security, and Rescue Robotics*. IEEE, pp. 327–332. DOI: 10.1109/SSRR.2011.6106751.
- Couceiro, Micael S., Rui P. Rocha, and Nuno M.F. Ferreira (2011b). "A Novel Multi-Robot Exploration Approach based on Particle Swarm Optimization Algorithms".

- In: *2011 IEEE International Symposium on Safety, Security, and Rescue Robotics*. Kyoto, Japan: IEEE, pp. 327–332. ISBN: 9781612847696. DOI: 10.1109/SSRR.2011.6106751.
- Couceiro, Micael S, Rui P Rocha, and Nuno MF Ferreira (2011c). “Ensuring ad hoc connectivity in distributed search with robotic darwinian particle swarms”. In: *2011 IEEE International Symposium on Safety, Security, and Rescue Robotics*. IEEE, pp. 284–289. DOI: 10.1109/SSRR.2011.6106752.
- Couceiro, Micael S. et al. (2014a). “Benchmark of swarm robotics distributed techniques in a search task”. In: *Robotics and Autonomous Systems* 62.2, pp. 200–213. ISSN: 09218890. DOI: 10.1016/j.robot.2013.10.004.
- Couceiro, Micael S et al. (2014b). “Darwinian swarm exploration under communication constraints: Initial deployment and fault-tolerance assessment”. In: *Robotics and Autonomous Systems* 62.4, pp. 528–544. DOI: 10.1016/j.robot.2013.12.009.
- Crosscombe, Michael and Jonathan Lawry (2021). “The Impact of Network Connectivity on Collective Learning”. In: *15th International Symposium on Distributed Autonomous Robotics Systems*. eprint: 2106.00655.
- Czirók, András, H. Eugene Stanley, and Tamás Vicsek (1997). “Spontaneously ordered motion of self-propelled particles”. In: *J. Phys A:Math. Gen* 30, p. 1375. DOI: 10.1088/0305-4470/30/5/009.
- Dadgar, Masoud, Micael S. Couceiro, and Ali Hamzeh (2017). “RDPSO diversity enhancement based on repulsion between similar ions for robotic target searching”. In: *2017 Artificial Intelligence and Signal Processing Conference (AISP)*. Shiraz, Iran: IEEE, pp. 275–280. ISBN: 9781538625859. DOI: 10.1109/AISP.2017.8324096.
- (2020). “RbRDPSO: repulsion-based RDPSO for robotic target searching”. In: *Iranian Journal of Science and Technology - Transactions of Electrical Engineering* 44.1, pp. 551–563. ISSN: 23641827. DOI: 10.1007/s40998-019-00245-z. URL: <https://doi.org/10.1007/s40998-019-00245-z>.
- Dadgar, Masoud, Shahram Jafari, and Ali Hamzeh (2016). “A PSO-based multi-robot cooperation method for target searching in unknown environments”. In: *Neurocomputing* 177, pp. 62–74. ISSN: 18728286. DOI: 10.1016/j.neucom.2015.11.007.
- De Oca, Marco A Montes et al. (2009). “Frankenstein’s PSO: a composite particle swarm optimization algorithm”. In: *IEEE Transactions on Evolutionary Computation* 13.5, pp. 1120–1132. DOI: 10.1109/TEVC.2009.2021465.
- Dorigo, Marco (1992). “Optimization, Learning and Natural Algorithms”. PhD Thesis. Politecnico di Milano.
- Dorigo, Marco, Mauro Birattari, and Thomas Stutzle (2006). “Ant colony optimization”. In: *IEEE computational intelligence magazine* 1.4, pp. 28–39. DOI: 10.1109/MCI.2006.329691.
- Dorigo, Marco, Guy Theraulaz, and Vito Trianni (2021). “Swarm Robotics: Past, Present, and Future”. In: *Proceedings of the IEEE* 109.7, pp. 1152–1165. ISSN: 0018-9219. DOI: 10.1109/jproc.2021.3072740.
- Dorigo, Marco et al. (2013). “Swarmanoid: A novel concept for the study of heterogeneous robotic swarms”. In: *IEEE Robotics and Automation Magazine* 20 (4), pp. 60–71. ISSN: 10709932. DOI: 10.1109/MRA.2013.2252996.

- Duarte, Miguel et al. (2016). "Evolution of Collective Behaviors for a Real Swarm of Aquatic Surface Robots". In: *PLoS ONE* 11.3. DOI: 10.1371/journal.pone.0151834.
- Eberhart, Russell and Yuhui Shi (2001). "Tracking and Optimizing Dynamic Systems with Particle Swarms". In: *Proceedings of the 2001 Congress on Evolutionary Computation*. Seoul, South Korea: IEEE, pp. 94–100. DOI: 10.1109/cec.2001.934376.
- Ebert, Julia T et al. (2020). "Bayes Bots : Collective Bayesian Decision-Making in Decentralized Robot Swarms". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. Paris, France, pp. 7186–7192. ISBN: 9781728173955.
- Elamvazhuthi, Karthik and Spring Berman (2019). "Mean-field models in swarm robotics: A survey". In: *Bioinspiration & Biomimetics* 15.015001. DOI: 10.1088/1748-3190/ab49a4.
- Engelbrecht, Andries P (2010). "Heterogeneous Particle Swarm Optimization". In: *7th Int. Conf. ANTS 2010*. Ed. by Dorigo, M. et al. Brussels, Belgium: Springer, Berlin, Heidelberg, pp. 191–202.
- Esterle, Lukas (2018a). "CHAINMAIL: Distributed coordination for multi-task -Assignment using autonomous mobile IoT devices". In: *Institute of Electrical and Electronics Engineers Inc.*, pp. 85–92. ISBN: 9781538654705. DOI: 10.1109/DCOSS.2018.00019.
- (2018b). "Goal-aware team affiliation in collectives of autonomous robots". In: pp. 90–99. ISBN: 9781538651728. DOI: 10.1109/SASO.2018.00020.
- Esterle, Lukas and David W King (2022). "Loosening Control — A Hybrid Approach to Controlling Heterogeneous Swarms". In: *ACM Transactions on Autonomous and Adaptive Systems* 16.2. DOI: 10.1145/3502725.
- Esterle, Lukas and Peter R Lewis (2017). "Online multi-object k-coverage with mobile smart cameras". In: *Proceedings of the 11th International Conference on Distributed Smart Cameras*, pp. 107–112. DOI: 10.1145/3131885.3131909.
- Esterle, Lukas and Peter R. Lewis (2020). "Distributed autonomy and trade-offs in on-line multiobject k-coverage". In: *Computational Intelligence* 36.2, pp. 720–742. ISSN: 14678640. DOI: 10.1111/coin.12264.
- Falcón-Cortés, Andrea, Denis Boyer, and Gabriel Ramos-Fernández (2019). "Collective learning from individual experiences and information transfer during group foraging". In: *Journal of the Royal Society Interface* 16.151. ISSN: 17425662. DOI: 10.1098/rsif.2018.0803. eprint: 1901.07465.
- Fang, Xu, Cheng Cheng, and Lihua Xie (2020). "3-D Multi-player Pursuit-evasion Game with a Faster Evader". In: *2020 39th Chinese Control Conference (CCC)*. IEEE. Shenyang, China, pp. 118–123. DOI: 10.23919/CCC50068.2020.9188950.
- Farid, Ali Moltajaei et al. (2018). "Search and tracking in 3D space using a species based particle swarm optimizer". In: *Proceedings - 2018 IEEE International Conference on Automatic Control and Intelligent Systems, I2CACIS 2018*. Shah Alam, Malaysia: IEEE, pp. 155–160. ISBN: 9781538656549. DOI: 10.1109/I2CACIS.2018.8603709.
- Francesca, Gianpiero and Mauro Birattari (2016). "Automatic design of robot swarms: Achievements and challenges". In: *Frontiers Robotics AI* 3. ISSN: 22969144. DOI: 10.3389/frobt.2016.00029.
- Fraseri, Mirgita, Lukas Esterle, and Alessandro Vittorio Papadopoulos (2020). "Cooperative Multi-agent Systems for the Multi-target κ -Coverage Problem". In: *Proceedings of the 12th International Conference on Agents and Artificial Intelligence, ICAART*

2020. Valletta, Malta: Springer International Publishing, pp. 106–131. ISBN: 9783030711580. DOI: 10.1007/978-3-030-71158-0.
- Gazi, Veysel and Kevin M Passino (2004). “A class of attractions/repulsion functions for stable swarm aggregations”. In: *International Journal of Control* 77.18, pp. 1567–1579. DOI: 10.1080/00207170412331330021.
- Ge, Hengqing, Guibin Chen, and Guang Xu (2018). “Multi-AUV cooperative target hunting based on improved potential field in a surface-water environment”. In: *Applied Sciences* 8.6, p. 973. DOI: 10.3390/app8060973.
- Gomes, Jorge, Paulo Urbano, and Anders Lyhne Christensen (2013). “Evolution of swarm robotics systems with novelty search”. In: *Swarm Intelligence* 7.2-3, pp. 115–144. ISSN: 19353820. DOI: 10.1007/s11721-013-0081-z. eprint: 1304.3362.
- Grassé, Plerre-P (1959). “La reconstruction du nid et les coordinations interindividuelles chez *Bellicositermes natalensis* et *Cubitermes* sp. la théorie de la stigmergie: Essai d’interprétation du comportement des termites constructeurs”. In: *Insectes sociaux* 6.1, pp. 41–80.
- Gupta, Anil K, Ken G Smith, and Christina E Shalley (2006). “The interplay between exploration and exploitation”. In: *Academy of management journal* 49.4, pp. 693–706. DOI: 10.5465/AMJ.2006.22083026.
- Hamann, Heiko (2012). “Towards swarm calculus: Universal properties of swarm performance and collective decisions”. In: *Swarm Intelligence: 8th International Conference, ANTS 2012. Volume 7461 of LNCS*. Berlin, Germany: Springer, pp. 168–179. DOI: 10.1007/978-3-642-32650-9_15.
- (2018). “Superlinear scalability in parallel computing and multi-robot systems: Shared resources, collaboration, and network topology”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 10793 LNCS, pp. 31–42. ISSN: 16113349. DOI: 10.1007/978-3-319-77610-1_3.
- Hansen, Nikolaus (2006). “The CMA evolution strategy: a comparing review”. In: *Towards a new evolutionary computation*, pp. 75–102. DOI: 10.1007/3-540-32494-1_4.
- Hansen, Nikolaus and Andreas Ostermeier (2001). “Completely derandomized self-adaptation in evolution strategies”. In: *Evolutionary computation* 9.2, pp. 159–195. DOI: 10.1162/106365601750190398.
- Harikumar, K., J. Senthilnath, and Suresh Sundaram (2019). “Multi-UAV Oxyrrhis Marina-Inspired Search and Dynamic Formation Control for Forest Firefighting”. In: *IEEE Transactions on Automation Science and Engineering* 16.2, pp. 863–873. ISSN: 15455955. DOI: 10.1109/TASE.2018.2867614.
- Hayes, Adam T, Alcherio Martinoli, and Rodney M Goodman (2002). “Distributed odor source localization”. In: *IEEE Sensors Journal* 2.3, pp. 260–271. DOI: 10.1109/jsen.2002.800682.
- Hecker, Joshua P, Justin Craig Carmichael, and Melanie E Moses (2015). “Exploiting Clusters for Complete Resource Collection in Biologically-Inspired Robot Swarms”. In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) Congress*. Hamburg, Germany, pp. 434–440. DOI: 10.1109/iros.2015.7353409.

- Hereford, James M. and Michael A Siebold (2010). "Bio-inspired search strategies for robot swarms". In: *Swarm Robotics from Biology to Robotics*. Ed. by Ester Martinez Martin. IntechOpen. DOI: 10.5772/8600.
- Herianto and Daisuke Kurabayashi (2009). "Realization of an artificial pheromone system in random data carriers using RFID tags for autonomous navigation". In: *2009 IEEE International Conference on Robotics and Automation*. IEEE, pp. 2288–2293. DOI: 10.1109/ROBOT.2009.5152405.
- Herianto, Toshiki Sakakibara, and Daisuke Kurabayashi (2007). "Artificial pheromone system using RFID for navigation of autonomous robots". In: *Journal of Bionic Engineering* 4.4, pp. 245–253. DOI: 10.1016/S1672-6529(07)60038-9.
- Hornischer, Hannes et al. (2020). "CIMAX: Collective Information Maximization in Robotic Swarms Using Local Communication". In: *Adaptive Behavior*. DOI: 10.1177/1059712320912021. eprint: 1903.05444.
- Horsevad, Nikolaj (2021). "Understanding Collective Behaviors Through Network Topology". PhD Thesis. Singapore University of Technology and Design.
- Horsevad, Nikolaj, Hian Lee Kwa, and Roland Bouffanais (2022). "Beyond Bio-Inspired Robotics: How Multi-Robot Systems Can Support Research on Collective Animal Behavior". In: *Frontiers in Robotics and AI* 9 (865414). DOI: 10.3389/frobt.2022.865414.
- Horsevad, Nikolaj et al. (2022). "Transition from simple to complex contagion in collective decision-making". In: *Nature Communications* 13.1442. DOI: 10.1038/s41467-022-28958-6.
- Hu, Xiaohui and Russell C. Eberhart (2002). "Adaptive particle swarm optimization: Detection and response to dynamic systems". In: *Proceedings of the 2002 Congress on Evolutionary Computation, CEC 2002*. Vol. 2. Honolulu, HI, USA, pp. 1666–1670. ISBN: 0780372824. DOI: 10.1109/CEC.2002.1004492.
- Huang, Chi Fu and Yu Chee Tseng (2005). "The coverage problem in a wireless sensor network". In: *Mobile Networks and Applications* 10.4, pp. 519–528. ISSN: 1383469X. DOI: 10.1007/s11036-005-1564-y.
- Hussain, Kashif et al. (2019). "On the exploration and exploitation in popular swarm-based metaheuristic algorithms". In: *Neural Computing and Applications* 31.11, pp. 7665–7683. DOI: 10.1007/s00521-018-3592-0.
- Hüttenrauch, Maximilian, Adrian Oic, and Gerhard Neumann (2019). "Deep reinforcement learning for swarm systems". In: *Journal of Machine Learning Research* 20, pp. 1–31. ISSN: 15337928. eprint: 1807.06613.
- Hüttenrauch, Maximilian, Adrian Šošić, and Gerhard Neumann (2018). "Local communication protocols for learning complex swarm behaviors with deep reinforcement learning". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 11172 LNCS, pp. 71–83. ISSN: 16113349. DOI: 10.1007/978-3-030-00533-7_6. eprint: 1709.07224.
- Janosov, Milán et al. (2017). "Group chasing tactics: How to catch a faster prey". In: *New Journal of Physics* 19.5. ISSN: 13672630. DOI: 10.1088/1367-2630/aa69e7. eprint: 1701.00284.
- Janson, Stefan and Martin Middendorf (2005). "A hierarchical particle swarm optimizer and its adaptive variant". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 35.6, pp. 1272–1282. DOI: 10.1109/TSMCB.2005.850530.

- Jatmiko, Wisnu, Kosuke Sekiyama, and Toshio Fukuda (2007). "A PSO-Based Mobile Robot for Odor Source Localization in Dynamic Advection-Diffusion with Obstacles Environment: Theory, Simulation and Measurement". In: *IEEE Computational Intelligence Magazine* 2.2, pp. 37–51. DOI: 10.1109/MCI.2007.353419.
- Johansson, Robert and Alessandro Saffiotti (2009). "Navigating by stigmergy: A realization on an RFID floor for minimalistic robots". In: *2009 IEEE International Conference on Robotics and Automation*. IEEE, pp. 245–252. DOI: 10.1109/robot.2009.5152737.
- Jordehi, Ahmad Rezaee (2014). "Particle Swarm Optimisation for Dynamic Optimisation Problems: A Review". In: *Neural Computing and Applications* 25, pp. 1507–1516. DOI: 10.1007/s00521-014-1661-6.
- Jurt, Marius et al. (2022). "Collective transport of arbitrarily shaped objects using robot swarms". In: *Artificial Life and Robotics*. ISSN: 1433-5298. DOI: 10.1007/s10015-022-00730-5. URL: <https://doi.org/10.1007/s10015-022-00730-5>.
- Kamimura, Atsushi, Shigenori Masumoto, and Toru Ohira (2015). "Mathematical approaches to biological systems: Networks, oscillations, and collective motions". In: *Mathematical Approaches to Biological Systems: Networks, Oscillations, and Collective Motions*. Ed. by Toru Ohira and Tohru Uzawa. Springer Japan, pp. 139–166. DOI: 10.1007/978-4-431-55444-8.
- Kamimura, Atsushi and Toru Ohira (2019). "Fusion of Pursuits-Escapes and Collective Motions". In: *Group Chase and Escape*. Springer Singapore, pp. 43–75. ISBN: 978-981-15-1731-0. DOI: 10.1007/978-981-15-1731-0_4. eprint: 0912.4327.
- Kegeleirs, Miquel, David Garzón Ramos, and Mauro Birattari (2019). "Random walk exploration for swarm mapping". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 11650 LNAI, pp. 211–222. ISSN: 16113349. DOI: 10.1007/978-3-030-25332-5_19.
- Kengyel, Daniela et al. (2016). "Collective decision making in a swarm of robots: How robust the BEECLUST algorithm performs in various conditions". In: *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS)*, pp. 264–271. DOI: 10.4108/eai.3-12-2015.2262332.
- Kennedy, James and Russell Eberhart (1995). "Particle Swarm Optimization". In: *Proceedings of ICNN'95 - International Conference on Neural Networks*. Perth, WA, Australia: IEEE, pp. 1942–1948. ISBN: 0780327683. DOI: 10.1109/ICNN.1995.488968.
- Kernbach, Serge et al. (2009). "Re-embodiment of honeybee aggregation behavior in an artificial micro-robotic system". In: *Adaptive Behavior* 17.3, pp. 237–259. ISSN: 17412633. DOI: 10.1177/1059712309104966.
- Khaliq, Ali Abdul (2018). "From Ants to Service Robots: an Exploration in Stigmergy-Based Navigation Algorithms". PhD Thesis. Örebro University. ISBN: 978-91-7529-253-3.
- Khaliq, Ali Abdul, Maurizio Di Rocco, and Alessandro Saffiotti (2014). "Stigmergic algorithms for multiple minimalistic robots on an RFID floor". In: *Swarm Intelligence* 8.3, pp. 199–225. ISSN: 19353820. DOI: 10.1007/s11721-014-0096-0.
- Khan, Asif, Bernhard Rinner, and Andrea Cavallaro (2018). "Cooperative robots to observe moving targets: Review". In: *IEEE Transactions on Cybernetics* 48.1, pp. 187–198. ISSN: 21682267. DOI: 10.1109/TCYB.2016.2628161.

- Kit, Jabez Leong (2021). "Towards a Swarm Intelligence Benchmark Framework". PhD Thesis. Singapore University of Technology and Design.
- Kit, Jabez Leong et al. (2019). "Decentralized Multi-Floor Exploration by a Swarm of Miniature Robots Teaming with Wall-Climbing Units". In: *2019 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*. New Brunswick, NJ, USA: IEEE. DOI: 10.1109/MRS.2019.8901058.
- Kolling, Andreas and Stefano Carpin (2006). "Multirobot cooperation for surveillance of multiple moving targets - A new behavioral approach". In: *Proceedings - IEEE International Conference on Robotics and Automation*. Orlando, FL: IEEE, pp. 1311–1316. ISBN: 0780395069. DOI: 10.1109/ROBOT.2006.1641890.
- (2007). "Cooperative observation of multiple moving targets: an algorithm and its formalization". In: *The International Journal of Robotics Research* 26.9, pp. 935–953. DOI: 10.1177/0278364907080424.
- Kolling, Andreas et al. (2016). "Human Interaction with Robot Swarms: A Survey". In: *IEEE Transactions on Human-Machine Systems* 46.1, pp. 9–26. ISSN: 21682291. DOI: 10.1109/THMS.2015.2480801.
- Komareji, Mohammad and Roland Bouffanais (2013). "Resilience and controllability of dynamic collective behaviors". In: *PLoS ONE* 8.12. ISSN: 19326203. DOI: 10.1371/journal.pone.0082578. arXiv: 1401.2598.
- Kouzehgar, Maryam, Malika Meghiani, and Roland Bouffanais (2020). "Multi-Agent Reinforcement Learning for Dynamic Ocean Monitoring by a Swarm of Buoys". In: *Global OCEANS 2020: Singapore-U.S Gulf Coast*. DOI: 10.1109/IEEECONF38699.2020.9389128. eprint: 2012.11641.
- Kuan, Yoke Kong (2018). "Design of an effective swarming system for the pervasive monitoring of aquatic environments". PhD Thesis. Singapore: Singapore University of Technology and Design.
- Kwa, Hian Lee and Roland Bouffanais (2022). "The Effect of Network Connectivity on Exploration and Exploitation During Decentralized Collective Learning". In: URL: http://abmhub.cs.ucl.ac.uk/2022/camera_ready/Kwa_Bouffanais.pdf.
- Kwa, Hian Lee, Jabez Leong Kit, and Roland Bouffanais (2020). "Optimal Swarm Strategy for Dynamic Target Search and Tracking". In: pp. 672–680. DOI: 10.5555/3398761.3398842.
- (2021a). "Tailoring Exploration and Exploitation in Multi-Agent Systems with Short-Term Memory and Limited Social Interaction". In: URL: http://abmhub.cs.ucl.ac.uk/2021/camera_ready/Kwa_etal.pdf.
- (2021b). "Tracking Multiple Fast Targets With Swarms : Interplay Between Social Interaction and Agent Memory". In: DOI: 10.1162/isal_a_00376.
- (2022). "Balancing Collective Exploration and Exploitation in Multi-Agent and Multi-Robot Systems: A Review". In: *Frontiers in Robotics and AI* 8 (771520). DOI: 10.3389/frobt.2021.771520.
- Kwa, Hian Lee et al. (2020). "Heterogeneous Swarms for Maritime Dynamic Target Search and Tracking". In: IEEE. DOI: 10.1109/IEEECONF38699.2020.9389145.
- Kwa, Hian Lee et al. (2022). "Adapting the Exploration-Exploitation Balance in Heterogeneous Swarms: Tracking Evasive Targets". In: *Artificial Life*.

- Lawry, Jonathan (2022). "Heterogeneity and Robustness in Social Learning". In: DOI: 10.1162/isal_a_00480.
- Leonard, Barend J, Andries P Engelbrecht, and Andrich B van Wyk (2011). "Heterogeneous Particle Swarms in Dynamic Environments". In: *2011 IEEE Symposium on Swarm Intelligence*. Paris, France: IEEE. ISBN: 9781612840529. DOI: 10.1109/SIS.2011.5952564.
- Lerman, Kristina and Aram Galstyan (2001). "Mathematical Model of Foraging in a Group of Robots: Effect of Interference". In: *Autonomous Robots* 13.2, pp. 127–141. DOI: 10.1023/A:1019633424543.
- Li, Hanjun et al. (2017). "Decentralized stochastic control of robotic swarm density: Theory, simulation, and experiment". In: *IEEE International Conference on Intelligent Robots and Systems*. Vancouver, BC, Canada, pp. 4341–4347. ISBN: 9781538626825. DOI: 10.1109/IROS.2017.8206299.
- Li, Jiaoyang et al. (2020). "Moving Agents in Formation in Congested Environments". In: *Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020)*. Auckland, New Zealand: IFAAMAS, pp. 726–734. DOI: 10.5555/3398761.3398848.
- Li, Shoutao et al. (2014). "A hybrid search algorithm for swarm robots searching in an unknown environment". In: *PLoS ONE* 9.11, e111970. ISSN: 19326203. DOI: 10.1371/journal.pone.0111970.
- Liu, Qunfeng et al. (2016). "Topology Selection for Particle Swarm Optimization". In: *Information Sciences* 363. DOI: 10.1016/j.ins.2016.04.050.
- Liu, Ruochen et al. (2017). "A coevolutionary technique based on multi-swarm particle swarm optimization for dynamic multi-objective optimization". In: *European Journal of Operational Research* 261.3, pp. 1028–1051. DOI: 10.1016/j.ejor.2017.03.048.
- Liu, Zixuan, Michael Crosscombe, and Jonathan Lawry (2021). "Imprecise Fusion Operators for Collective Learning". In: *ALIFE 2021: The 2021 Conference on Artificial Life*. DOI: 10.1162/isal_a_00407.
- López, Daniel Serrano et al. (2017). "Interoperability in a Heterogeneous Team of Search and Rescue Robots". In: *Search and Rescue Robotics - From Theory to Practice*, pp. 93–125. DOI: 10.5772/intechopen.69493.
- Lv, Jinfeng et al. (2016). "Maritime Static Target Search Based on Particle Swarm Algorithm". In: *Proceedings of SAI Intelligent Systems Conference (IntelliSys) 2016*. Ed. by Y. Bi, S. Kapoor, and R. Bhatia. Vol. 16. London, UK: Springer Cham, pp. 917–927. ISBN: 978-3-319-56993-2. DOI: 10.1007/978-3-319-56994-9. eprint: 1606.07283.
- Mateo, David, Yoke Kong Kuan, and Roland Bouffanais (2017). "Effect of Correlations in Swarms on Collective Response". In: *Scientific Reports* 7.10388. ISSN: 20452322. DOI: 10.1038/s41598-017-09830-w.
- Mateo, David et al. (2019). "Optimal Network Topology for Effective Collective Response". In: *Science Advances* 5.4, eaau0999. DOI: 10.1126/sciadv.aau099. eprint: 1807.04631.
- Matignon, Laetitia and Olivier Simonin (2018). "Multi-robot simultaneous coverage and mapping of complex scene - Comparison of different strategies: Robotics track". In: *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*. Vol. 1. Stockholm, Sweden, pp. 559–567. ISBN: 9781510868083.

- Medina, Jose et al. (2017). "NASA Swarmathon Search and Rescue". In: *30th Florida Conference on Recent Advances in Robotics*. Boca Raton, FL, USA.
- Meghjani, Malika, Sandeep Manjanna, and Gregory Dudek (2016). "Multi-target rendezvous search". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'16)*. Daejeon, South Korea. ISBN: 9781509037629. DOI: 10.1109/IROS.2016.7759404.
- Mehlhorn, Katja et al. (2015). "Unpacking the exploration-exploitation tradeoff: A synthesis of human and animal literatures". In: *Decision* 2.3, pp. 191–215. ISSN: 23259973. DOI: 10.1037/dec0000033.
- Meyer-Nieberg, Silja (2017). "Coordinating a Team of Searchers: of Ants, Swarms, and Slime Molds". In: *Proceedings of GECCO '17*. Berlin, Germany: ACM, pp. 27–34. DOI: 10.1145/3071178.3071277.
- Meyer-Nieberg, Silja et al. (2013). "Intercepting a target with sensor swarms". In: *Proceedings of the Annual Hawaii International Conference on System Sciences*. Wailea, HI, USA, pp. 1222–1230. ISBN: 9780769548920. DOI: 10.1109/HICSS.2013.281.
- Nauta, Johannes, Yara Khaluf, and Pieter Simoens (2020). "Hybrid foraging in patchy environments using spatial memory". In: *Journal of the Royal Society Interface* 17.166. DOI: 10.1098/rsif.2020.0026.
- Nauta, Johannes, Pieter Simoens, and Yara Khaluf (2020). "Memory Induced Aggregation in Collective Foraging". In: *ANTS2020, The 13th International Conference on Swarm Intelligence*. Ed. by Marco Dorigo et al. Barcelona, Spain: Springer International Publishing, pp. 176–189. ISBN: 9783030603762. DOI: 10.1007/978-3-030-60376-2_14.
- Nauta, Johannes et al. (2020). "Enhanced foraging in robot swarms using collective levy walks". In: *24th European Conference on Artificial Intelligence - ECAI 2020*. Santiago de Compostela, Spain. ISBN: 9781643681009.
- Navarro, Daniel J, Ben R Newell, and Christin Schulze (2016). "Learning and choosing in an uncertain world: An investigation of the explore–exploit dilemma in static and dynamic environments". In: *Cognitive psychology* 85, pp. 43–77. DOI: 10.1016/j.cogpsych.2016.01.001.
- Nedjah, Nadia and Luneque Silva Junior (2019). "Review of methodologies and tasks in swarm robotics towards standardization". In: *Swarm and Evolutionary Computation* 50.July, p. 100565. ISSN: 22106502. DOI: 10.1016/j.swevo.2019.100565.
- Ni, Jianjun and Simon X. Yang (2011). "Bioinspired neural network for real-time cooperative hunting by multirobots in unknown environments". In: *IEEE Transactions on Neural Networks* 22, pp. 2062–2077. ISSN: 10459227. DOI: 10.1109/TNN.2011.2169808.
- Ni, Jianjun et al. (2018). "An Improved DSA-Based Approach for Multi-AUV Cooperative Search". In: *Computational Intelligence and Neuroscience* 2018.2186574. DOI: 10.1155/2018/2186574.
- Oliveira, Marcos et al. (2017). "Better Exploration-Exploitation Pace, Better Swarm: Examining the Social Interactions". In: *2017 IEEE Latin American Conference on Computational Intelligence (LA-CCI)*. Arequipa, Peru: IEEE. ISBN: 9781538637340. DOI: 10.1109/LA-CCI.2017.8285712.

- Palmieri, Nunzia and Salvatore Marano (2016). "Discrete firefly algorithm for recruiting task in a swarm of robots". In: *Studies in Computational Intelligence*. Vol. 637, pp. 133–150. ISBN: 9783319302355. DOI: 10.1007/978-3-319-30235-5_7.
- Pang, Bao et al. (2019). "A swarm robotic exploration strategy based on an improved random walk method". In: *Journal of Robotics* 2019. DOI: 10.1155/2019/6914212.
- Parker, Lynne E. (2002). "Distributed algorithms for multi-robot observation of multiple moving targets". In: *Autonomous Robots* 12.3, pp. 231–255. ISSN: 09295593. DOI: 10.1023/A:1015256330750.
- Parker, Lynne E. and Brad A. Emmons (1997). "Cooperative multi-robot observation of multiple moving targets". In: *Proceedings - IEEE International Conference on Robotics and Automation*. Vol. 3. Albuquerque, NM, pp. 2082–2089. ISBN: 0780336127. DOI: 10.1109/robot.1997.619270.
- Perna, Andrea et al. (2012). "Individual rules for trail pattern formation in Argentine ants (*Linepithema humile*)". In: *PLoS Comput Biol* 8.7, e1002592. DOI: 10.1371/journal.pcbi.1002592.
- Piotrowski, Adam P., Jaroslaw J. Napiorkowski, and Agnieszka E. Piotrowska (2020). "Population size in Particle Swarm Optimization". In: *Swarm and Evolutionary Computation* 58, p. 100718. DOI: 10.1016/j.swevo.2020.100718. URL: <https://doi.org/10.1016/j.swevo.2020.100718>.
- Prasetyo, Judhi, Giulia De Masi, and Eliseo Ferrante (2019). "Collective decision making in dynamic environments". In: *Swarm Intelligence* 13, pp. 217–243. DOI: 10.1007/s11721-019-00169-8.
- Pugh, Jim and Alcherio Martinoli (2007). "Inspiring and Modeling Multi-Robot Search with Particle Swarm Optimization". In: *2007 IEEE Swarm Intelligence Symposium*. Honolulu, HI, USA: IEEE, pp. 332–339. ISBN: 1424407087. DOI: 10.1109/SIS.2007.367956.
- Qi, Jingtao et al. (2020). "Group Chase and Escape of Biological Groups Based on a Visual Perception-Decision-Propulsion Model". In: *IEEE Access* 8, pp. 160490–160499. ISSN: 21693536. DOI: 10.1109/ACCESS.2020.3021078.
- Qolomany, Basheer et al. (2017). "Parameters optimization of deep learning models using Particle swarm optimization". In: *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE, pp. 1285–1290. DOI: 10.1109/IWCMC.2017.7986470.
- Ramachandran, R. K., Z. Kakish, and S. Berman (2020). "Information Correlated Lévy Walk Exploration and Distributed Mapping Using a Swarm of Robots". In: *IEEE Transactions on Robotics* 36.5, pp. 1422–1441. DOI: 10.1109/TRO.2020.2991612.
- Ramachandran, Ragesh Kumar (2018). "Exploration, mapping and scalar field estimation using a swarm of resource-constrained robots". PhD Thesis. Arizona State University.
- Rausch, Ilja et al. (2019). "Coherent collective behaviour emerging from decentralised balancing of social feedback and noise". In: *Swarm Intelligence* 13, pp. 321–345. ISSN: 1935-3812. DOI: 10.1007/s11721-019-00173-y.
- Reynolds, Craig W (1987). "Flocks, herds and schools: A distributed behavioral model". In: *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pp. 25–34. DOI: 10.1145/37402.37406.

- Ritz, Fabian et al. (2021). "A Sustainable Ecosystem through Emergent Cooperation in Multi-Agent Reinforcement Learning". In: *ALIFE 2021: The 2021 Conference on Artificial Life*. DOI: 10.1162/isal_a_00399.
- Roeva, Olympia, Stefka Fidanova, and Marcin Paprzycki (2015). "Population size influence on the genetic and ant algorithms performance in case of cultivation process modeling". In: *Recent advances in computational optimization*. Springer, pp. 107–120. DOI: 10.1007/978-3-319-12631-9_7.
- Rosenfeld, Avi, Gal A. Kaminka, and Sarit Kraus (2006). "A study of scalability properties in robotic teams". In: *Coordination of Large-Scale Multiagent Systems*. Ed. by Paul Scerri, Régis Vincent, and Roger Mailler. Boston, MA, USA: Springer, pp. 27–51. DOI: 10.1007/0-387-27972-5_2.
- Rosenthal, Sara Brin et al. (2015). "Revealing the hidden networks of interaction in mobile animal groups allows prediction of complex behavioral contagion". In: *Proceedings of the National Academy of Sciences* 112.15, pp. 4690–4695.
- Rubenstein, Michael et al. (2014). "Kilobot: A low cost robot with scalable operations designed for collective behaviors". In: *Robotics and Autonomous Systems* 62.7, pp. 966–975. DOI: 10.1016/j.robot.2013.08.006.
- Saffre, F. et al. (2019). "Long-term memory-induced synchronisation can impair collective performance in congested systems". In: *Swarm Intelligence*, pp. 1–20. ISSN: 19353820. DOI: 10.1007/s11721-019-00164-z.
- Sánchez, Nicolàs D Griffiths, Patricia A Vargas, and Micael S Couceiro (2018). "A darwinian swarm robotics strategy applied to underwater exploration". In: *2018 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, pp. 1–6. DOI: 10.1109/CEC.2018.8477738.
- Sardinha, Hugo, Mauro Dragone, and Patricia A. Vargas (2020a). "Combining Lévy Walks and Flocking for Cooperative Surveillance Using Aerial Swarms". In: *17th European Conference on Multi-Agent Systems 2020*. Thessaloniki, Greece: Springer International Publishing. ISBN: 9783030664114. DOI: 10.1007/978-3-030-66412-1_15.
- Sardinha, Hugo, Mauro Dragone, and Patricia A Vargas (2020b). "Towards an Adaptive Lévy Walk Using Artificial Endocrine Systems". In: *ADAPTIVE 2020: The Twelfth International Conference on Adaptive and Self-Adaptive Systems and Applications*. Nice, France, pp. 116–121. ISBN: 9781612087818.
- Schaerf, T. M. et al. (2013). "Do small swarms have an advantage when house hunting? The effect of swarm size on nest-site selection by *Apis mellifera*". In: *Journal of the Royal Society Interface* 10.87. ISSN: 17425689. DOI: 10.1098/rsif.2013.0533.
- Schmickl, Thomas et al. (2009). "Get in touch: Cooperative decision making based on robot-to-robot collisions". In: *Autonomous Agents and Multi-Agent Systems* 18.1, pp. 133–155. ISSN: 13872532. DOI: 10.1007/s10458-008-9058-5.
- Schranz, Melanie and Torsten Andre (2018). "Towards resource-aware hybrid camera systems". In: *International Conference on Distributed Smart Cameras (ICDSC '18)*. Eindhoven, Netherlands. ISBN: 9781450365116. DOI: 10.1145/3243394.3243701.
- Schranz, Melanie et al. (2021). "Swarm Intelligence and cyber-physical systems: Concepts, challenges and future trends". In: *Swarm and Evolutionary Computation* 60.100762. ISSN: 22106502. DOI: 10.1016/j.swevo.2020.100762.

- Schroeder, Adam, Brian Trease, and Alessandro Arsie (2019). "Balancing robot swarm cost and interference effects by varying robot quantity and size". In: *Swarm Intelligence* 13.1, pp. 1–19. ISSN: 19353820. DOI: 10.1007/s11721-018-0161-1.
- Schumer, MA and Kenneth Steiglitz (1968). "Adaptive step size random search". In: *IEEE Transactions on Automatic Control* 13.3, pp. 270–276. DOI: 10.1109/TAC.1968.1098903.
- Senanayake, Madhubhashi et al. (2016). "Search and tracking algorithms for swarms of robots: A survey". In: *Robotics and Autonomous Systems* 75, pp. 422–434. ISSN: 09218890. DOI: 10.1016/j.robot.2015.08.010.
- Shang, Yilun and Roland Bouffanais (2014). "Consensus reaching in swarms ruled by a hybrid metric-topological distance". In: *The European Physical Journal B* 87.12, pp. 1–7. DOI: 10.1140/epjb/e2014-50094-4.
- Shi, Yunfei et al. (2020). "Adaptive Informative Sampling with Environment Partitioning for Heterogeneous Multi-Robot Systems". In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Las Vegas, NV, USA, pp. 11718–11723. ISBN: 9781728162119.
- Shishika, Daigo and Derek A. Paley (2019). "Mosquito-inspired distributed swarming and pursuit for cooperative defense against fast intruders". In: *Autonomous Robots* 43.7, pp. 1781–1799. ISSN: 15737527. DOI: 10.1007/s10514-018-09827-y.
- Sörensen, Kenneth (2015). "Metaheuristics-the metaphor exposed". In: *International Transactions in Operational Research* 22.1, pp. 3–18. ISSN: 14753995. DOI: 10.1111/itor.12001.
- Souza, C de, P Castillo, and B Vidolov (2022). "Local interaction and navigation guidance for hunters drones : A chase behavior approach with real-time tests". In: *Robotica*, pp. 1–19. DOI: 10.1017/S0263574721001910.
- Stadler, Christian, Tazeeb Rajwani, and Florence Karaba (2014). "Solutions to the exploration/exploitation dilemma: Networks as a new level of analysis". In: *International Journal of Management Reviews* 16.2, pp. 172–193. ISSN: 14682370. DOI: 10.1111/ijmr.12015.
- Stogiannos, Marios, Alex Alexandridis, and Haralambos Sarimveis (2020). "An enhanced decentralized artificial immune-based strategy formulation algorithm for swarms of autonomous vehicles". In: *Applied Soft Computing Journal* 89, p. 106135. ISSN: 15684946. DOI: 10.1016/j.asoc.2020.106135. URL: <https://doi.org/10.1016/j.asoc.2020.106135>.
- Strickland, Laura et al. (2018). "Bio-inspired Role Allocation of Heterogeneous Teams in A Site Defense Task". In: *Distributed Autonomous Robotic Systems 2018*. Boulder, CO, USA: Springer International Publishing. ISBN: 9783030058166. DOI: 10.1007/978-3-030-05816-6_10.
- Sumpter, David J. T. (2010). *Collective Animal Behavior*. Princeton University Press. ISBN: 978-0-691-12963-1.
- Sun, Zhiyuan et al. (2022). "Self-Organizing Cooperative Pursuit Strategy for Multi-USV with Dynamic Obstacle Ships". In: *Journal of Marine Science and Engineering* 10.562. DOI: 10.3390/jmse10050562.

- Sundram, Jugesh et al. (2019). "Development of a Miniature Robot for Multi-robot Occupancy Grid Mapping". In: *ICARM 2018 - 2018 3rd International Conference on Advanced Robotics and Mechatronics*. Singapore: IEEE, pp. 414–419. DOI: 10.1109/ICARM.2018.8610745.
- Sung, Yoonchang et al. (2018). "Distributed simultaneous action and target assignment for multi-robot multi-target tracking". In: *Proceedings - IEEE International Conference on Robotics and Automation*. Brisbane, Australia: IEEE, pp. 3724–3729. ISBN: 9781538630815. DOI: 10.1109/ICRA.2018.8460974. eprint: 1706.02245.
- (2020). "Distributed Assignment with Limited Communication for Multi-Robot Multi-Target Tracking". In: *Autonomous Robots* 44, pp. 57–73. ISSN: 1573-7527. DOI: 10.1007/s10514-019-09856-1. eprint: 1812.11172.
- Sutanty, Donny et al. (2013). "Collective-adaptive lévy flight for underwater multi-robot exploration". In: *2013 IEEE International Conference on Mechatronics and Automation*. IEEE. Takamatsu, Japan, pp. 456–462.
- Talamali, Mohamed S. et al. (2021). "When less is more: Robot swarms adapt better to changes with constrained communication". In: *Science Robotics* 6.56. ISSN: 24709476. DOI: 10.1126/scirobotics.abf1416.
- Tan, Yong Chye and Bradley E Bishop (2004). "Evaluation of robot swarm control methods for underwater mine countermeasures". In: *Proceedings of the Thirty-Sixth Southeastern Symposium on System Theory*. Atlanta, GA, USA, pp. 294–298. ISBN: 0-7803-8281-1. DOI: 10.1109/SSST.2004.1295667.
- Tang, Qirong et al. (2019). "Dynamic target searching and tracking with swarm robots based on stigmergy mechanism". In: *Robotics and Autonomous Systems* 120, p. 103251. ISSN: 09218890. DOI: 10.1016/j.robot.2019.103251.
- Tang, Qirong et al. (2020a). "A stigmergetic method based on vector pheromone for target search with swarm robots". In: *Journal of Experimental and Theoretical Artificial Intelligence* 32.3, pp. 533–555. ISSN: 13623079. DOI: 10.1080/0952813X.2019.1653384.
- Tang, Qirong et al. (2020b). "Swarm Robots Search for Multiple Targets". In: *IEEE Access* 8, pp. 92814–92826. DOI: 10.1109/ACCESS.2020.2994151.
- Thenius, Ronald et al. (2016). "subCULTron - Cultural development as a tool in underwater robotics". In: *Artificial Life and Intelligent Agents*. Vol. 732. Birmingham, UK: Springer International Publishing, pp. 27–41. ISBN: 9783319904177. DOI: 10.1007/978-3-319-90418-4_3.
- Theraulaz, Guy and Eric Bonabeau (1999). "A Brief History of Stigmergy". In: *Artificial Life* 5.2, pp. 97–116. ISSN: 1064-5462. DOI: 10.1162/106454699568700.
- Tillett, Jason et al. (2005). "Darwinian particle swarm optimization". In: *Proceedings of the 2nd Indian International Conference on Artificial Intelligence*. Pune, India.
- Tversky, Amos and Ward Edwards (1966). "Information versus reward in binary choices." In: *Journal of Experimental Psychology* 71.5, p. 680. DOI: 10.1037/h0023123.
- Vallegra, Francesco et al. (2018). "Gradual Collective Upgrade of a Swarm of Autonomous Buoys for Dynamic Ocean Monitoring". In: *IEEE-MTS OCEANS 2018*. Charleston, SC, USA. DOI: 10.1109/OCEANS.2018.8604642.

- Van Den Bergh, Frans and Andries P Engelbrecht (2001). "Effects of swarm size on cooperative particle swarm optimisers". In: *GECCO'01: Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*. San Fransisco, CA, USA, pp. 892–899. DOI: 10.5555/2955239.2955400.
- Vicsek, Tamás et al. (1995). "Novel Type of Phase Transition in a System of Self-Driven Particles". In: *Physical Review Letters* 75.6, pp. 132–135. DOI: 10.1103/PhysRevLett.75.1226.
- Viswanathan, G. M., E. P. Raposo, and M. G.E. da Luz (2008). "Lévy flights and superdiffusion in the context of biological encounters and random searches". In: *Physics of Life Reviews* 5.3, pp. 133–150. ISSN: 15710645. DOI: 10.1016/j.plrev.2008.03.002.
- Viswanathan, G M et al. (2000). "Lévy in random searches". In: *Physica A* 282, pp. 1–12. ISSN: 03784371. DOI: 10.1518/001872008X250656.
- Wahby, Mostafa et al. (2019). "Collective change detection: Adaptivity to dynamic swarm densities and light conditions in robot swarms". In: *Artificial Life Conference Proceedings*. MIT Press. Newcastle, United Kingdom, pp. 642–649. DOI: 10.1162/isal_a_00233.
- Wang, Jianan et al. (2022). "Pursuit-Evasion Games of Multiple Cooperative Pursuers and an Evader: A Biological-Inspired Perspective". In: *Communications in Nonlinear Science and Numerical Simulation*, p. 106386. ISSN: 1007-5704. DOI: 10.1016/j.cnsns.2022.106386.
- Wang, Ning et al. (2021). "Cooperative Target Search of UAV Swarm with Communication Distance Constraint". In: *Mathematical Problems in Engineering*, p. 3794329. ISSN: 15635147. DOI: 10.1155/2021/3794329.
- Wang, Pamela and Malika Meghjani (2020). "Lost at Sea: Multi-Searcher Multi-Target Search". In: *Global Oceans 2020: Singapore – U.S. Gulf Coast*. IEEE. ISBN: 9781728154466. DOI: 10.1109/IEEECONF38699.2020.9389479.
- Wang, Yiming and Andrea Cavallaro (2016). "Prioritized target tracking with active collaborative cameras". In: *2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2016*. Colorado Springs, CO, USA: IEEE, pp. 131–137. ISBN: 9781509038114. DOI: 10.1109/AVSS.2016.7738066.
- Ward, Ashley and Mike Webster (2016). "Attraction, Alignment and Repulsion: How Groups Form and How They Function." In: *Sociality: The Behaviour of Group-Living Animals*. Springer Cham. ISBN: 9783319285856. DOI: 10.1007/978-3-319-28585-6.
- Wolek, Artur et al. (2020). "Cooperative Mapping and Target Search over an Unknown Occupancy Graph Using Mutual Information". In: *IEEE Robotics and Automation Letters* 5.2, pp. 1071–1078. ISSN: 23773766. DOI: 10.1109/LRA.2020.2966394.
- Yan, Peng, Tao Jia, and Chengchao Bai (2021). "Searching and tracking an unknown number of targets: A learning-based method enhanced with maps merging". In: *Sensors* 21.1076. ISSN: 14248220. DOI: 10.3390/s21041076.
- Yan, Yuting et al. (2018). "Modified PSO algorithms with "Request and Reset" for leak source localization using multiple robots". In: *Neurocomputing* 292, pp. 82–90. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2018.02.078.

- Yang, Jian, Xin Wang, and Peter Bauer (2019). "Extended PSO Based Collaborative Searching for Robotic Swarms With Practical Constraints". In: *IEEE Access* 7, pp. 76328–76341. DOI: 10.1109/access.2019.2921621.
- Yang, Xueming et al. (2007). "A modified particle swarm optimizer with dynamic adaptation". In: *Applied Mathematics and Computation* 189.2, pp. 1205–1213. DOI: 10.1016/j.amc.2006.12.045.
- Zedadra, Ouarda et al. (2017). "Multi-Agent Foraging: state-of-the-art and research challenges". In: *Complex Adaptive Systems Modeling* 5.1. ISSN: 21943206. DOI: 10.1186/s40294-016-0041-8.
- Zedadra, Ouarda et al. (2018). "Lévy walk-based search strategy: Application to destructive foraging". In: *Proceedings of the 2018 13th International Symposium on Programming and Systems, ISPS 2018*. Algiers, Algeria: IEEE. ISBN: 9781538646908. DOI: 10.1109/ISPS.2018.8379010.
- Zedadra, Ouarda et al. (2019). "A lévy walk and firefly based multi-robots foraging algorithm". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 11874 LNCS. Springer International Publishing, pp. 213–222. ISBN: 9783030349134. DOI: 10.1007/978-3-030-34914-1_21. URL: http://dx.doi.org/10.1007/978-3-030-34914-1_{_}21.
- Zhan, Zhi-hui et al. (2009). "Adaptive Particle Swarm Optimization". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 39.6, pp. 1362–1381. DOI: 10.1109/TSMCB.2009.2015956.
- Zhang, Guo You, Jian Chao Zeng, and Song Dong Xue (2014). "Research on task allocation of multi-target search with swarm robots". In: *International Journal of Wireless and Mobile Computing* 7.3, pp. 297–304. ISSN: 17411092. DOI: 10.1504/IJWMC.2014.062039.
- Zhang, S. et al. (2019a). "Dynamics and motion patterns of a k-capture game with attraction-repulsion interaction". In: *EPL (Europhysics Letters)* 128.10003. ISSN: 12864854. DOI: 10.1209/0295-5075/128/10003.
- (2019b). "Dynamics and motion patterns of a k-capture game with attraction-repulsion interaction". In: *EPL (Europhysics Letters)* 128.10003. ISSN: 12864854. DOI: 10.1209/0295-5075/128/10003.
- Zhong, Vivienne Jia 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)*. Singapore: IEEE, pp. 160–164. ISBN: 9781538663318. DOI: 10.1109/ICoIAS.2018.8494196.
- Zoss, Brandon M. et al. (2018). "Distributed system of autonomous buoys for scalable deployment and monitoring of large waterbodies". In: *Autonomous Robots* 42.8, pp. 1669–1689. ISSN: 15737527. DOI: 10.1007/s10514-018-9702-0.
- Şahin, Erol (2005). *Swarm Robotics: From Sources of Inspiration to Domains of Application*. Springer Berlin Heidelberg, pp. 10–20. DOI: 10.1007/978-3-540-30552-1_2.