

# Swarm Robotics

Foraging in cluttered environments:  
An analysis of non-stigmergic approaches

**Daniel Alexander Mead**

Supervisor:  
Prof. Stephen Hailes

MSc in Robotics and Computation  
September 2019

This report is submitted as part requirement for the MSc in Robotics and Computation at University College London. It is substantially the result of my own work except where explicitly indicated in the text.

The report may be freely copied and distributed provided the source is explicitly acknowledged.

Department of Computer Science  
University College London



# Abstract

Foraging is a commonly examined behaviour in swarm robotics, with cluttered and foreign environments posing additional problems to the development of an optimal solution. Many current solutions rely on hardware- or context-dependent stigmergic methods which may not be suitable for environments outside the laboratory.

Five non-stigmergic foraging behaviours were implemented onto a swarm of foot-bots. Four of these were based on existing algorithms, with the fifth utilising characteristics from the others, as well as some novel features. These were tested in environments of varying design and clutter-levels, quantified by their fractal dimension, and their relative performance recorded. The high dimensionality of the system prevented an in-depth implementation of a set of optimally-tuned parameters. Performance variations from using optimal parameters vs. a standard profile were found to be less than those attributable to swarm size, environment type, or foraging behaviour.

The results found the interaction-mediated behaviour to be inferior to all others, random walks to be the most effective only at very small swarm sizes in low-clutter environments, and information-mediated behaviour to be superior in all other situations. The new method exhibited the best performance at low-to-medium swarm sizes, and competitive results thereafter, identifying it as a target for further development.



# Acknowledgements

To my family for their unconditional support and love in all I do; for showing pride in my successes and sympathy in my setbacks.

To my supervisor Steve Hailes for his guidance, patience, and imparting his seemingly boundless wisdom and experience where it was needed.

To the Robotics Cohort for sharing in a truly memorable year; proof that adversity can be tolerated, and even enjoyed, with good friends around you.

To my best friend and co-pilot, Olly. You have been a source of endless encouragement and have never lost faith in me. This year would have been so much less for the lack of you. I hope for many more. I love you.

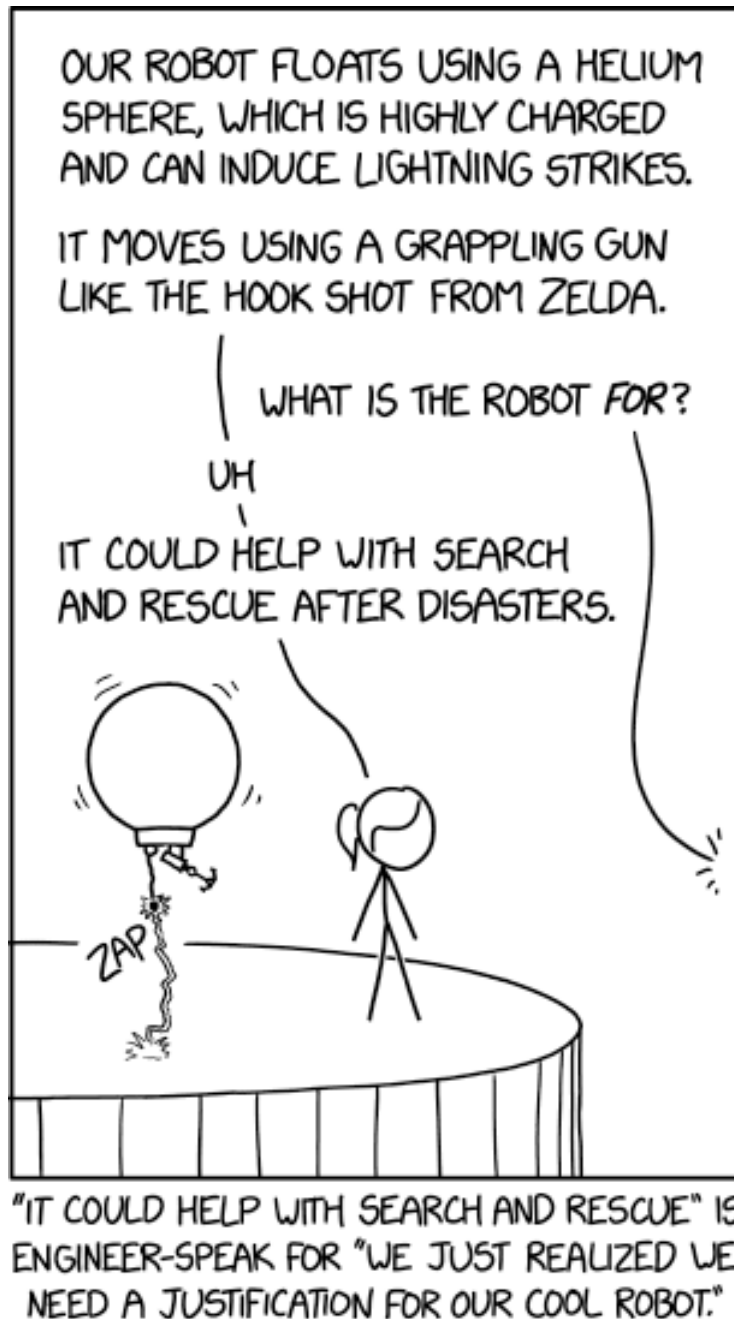


‘Go to the ant, thou sluggard; consider her ways, and be wise:  
Which having no guide, overseer, or ruler,  
Provideth her meat in the summer, and gathereth her food in the harvest.’

- Proverbs 6:6-8

‘ANTS! ANTS!’

- Michael Douglas, ‘*Antman*’, 2015



**Figure 1:** Source: Randall Munroe 'New Robot', 2019 [1]  
(This strikes a little too true.)



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Background</b>	<b>5</b>
2.1	Swarm Theory . . . . .	5
2.1.1	Swarm Robotics . . . . .	5
2.1.2	Biological inspiration . . . . .	6
2.2	Swarm Mechanics . . . . .	9
2.2.1	Emergent Behaviour . . . . .	9
2.2.2	Development Methods . . . . .	10
2.2.3	Communication . . . . .	12
2.3	Foraging . . . . .	14
2.3.1	Cluttered Environments . . . . .	15
2.3.2	Stigmergic Methods . . . . .	15
2.3.3	Waggle Dance . . . . .	17
2.3.4	Other non-stigmergic methods . . . . .	18
2.3.5	Role of Errors / Randomness . . . . .	21
<b>3</b>	<b>Methods</b>	<b>23</b>
3.1	Environment . . . . .	23
3.2	Behavioural Models . . . . .	26
3.2.1	Foraging . . . . .	26
3.2.2	Correlated Random Walk . . . . .	26
3.2.3	Site Fidelity (Biased Random Walk) . . . . .	28
3.2.4	Ducatelle Method (Information-mediated behaviour) . . . . .	29
3.2.5	Kasprzok Method (Encounter-mediated behaviour) . . . . .	30
3.2.6	Composite Method . . . . .	33
3.2.7	Tuning . . . . .	36
3.2.8	System Architecture . . . . .	38

<b>4</b>	<b>Results</b>	<b>41</b>
4.1	Analytical Results . . . . .	41
4.2	Arena Comparisons . . . . .	42
4.2.1	Arena 1 (Open Arena, $D_b = 1.14$ ) . . . . .	43
4.2.2	Arena 2 (Low Clutter, $D_b = 1.16$ ) . . . . .	46
4.2.3	Arena 3 (High Clutter, $D_b = 1.29$ ) . . . . .	48
4.2.4	Arena 4 (Maze, $D_b = 1.25$ ) . . . . .	49
4.2.5	Random Arenas ( $D_b = 1.473$ ( $\sigma = 0.008$ )) . . . . .	50
4.2.6	Overall Observations . . . . .	51
4.3	Changes in optimal parameters . . . . .	52
<b>5</b>	<b>Conclusions</b>	<b>57</b>
5.1	Optimal Parameters in Foraging . . . . .	57
5.2	Algorithm Comparisons . . . . .	58
5.3	Composite Method . . . . .	59
5.4	Improvements and Future Work . . . . .	60
5.5	Summary . . . . .	62

# List of Figures

1	XKCD: ‘ <i>New Robot</i> ’ . . . . .	viii
3.1	Static arena designs. . . . .	24
3.2	Random arena designs. . . . .	24
3.3	Von Mises distributions with varying values of $\kappa$ with $\mu$ set to 0. . . . .	27
3.4	Combined von Mises distributions. . . . .	28
3.5	Ducatelle method broadcast format. . . . .	31
3.6	Composite method broadcast format. . . . .	34
3.7	Parameter independence results. . . . .	38
3.8	System Architecture . . . . .	39
4.1	Ducatelle method in progress, low clutter. . . . .	41
4.2	Ducatelle method in progress, high clutter. . . . .	42
4.3	Arena 1 results; standard setup. . . . .	43
4.4	Arena 1 results; uniform initial. . . . .	43
4.5	Arena 1 results; offset landmarks. . . . .	44
4.6	Arena 2 results; standard setup. . . . .	46
4.7	Arena 2 results; uniform initial. . . . .	46
4.8	Arena 2 results; offset landmarks. . . . .	47
4.9	Arena 3 results. . . . .	48
4.10	Ducatelle method trials. . . . .	49
4.11	Arena 4 results. . . . .	49
4.12	Random arena results. . . . .	50
4.13	Variation in optimal parameters. . . . .	54



# List of Tables

3.1	Optimal parameter values obtained through tuning. . . . .	39
4.1	Average residuals from optimal parameter values against the ‘standard profile’. . . . .	54



# Chapter 1

## Introduction

Swarm robotics has evolved steadily as a distinct field of robotic control systems. Where classical robotics systems tend towards individual agents with increased capabilities via expensive hardware and complex control algorithms, swarm robotics seeks to accomplish tasks using the inverse approach, leveraging the capabilities of large numbers of operators to achieve goals, in spite of simplistic design and construction. Robotic swarms take advantage of decreasing costs in hardware and increases in computing power to generate solutions that cannot always be achieved as effectively, cheaply, or reliably with a single operator.

The primary barrier to effective swarm solutions is that of control: where individual capabilities are limited, designs must rely on agent interactions to accomplish tasks. While individual behaviours may be easy to design, it is significantly harder to develop solutions invoking the use of the entire swarm in a manner that is predictable and controllable. The field of swarm robotics is growing rapidly, but still lacks a degree of centralised design and formalism; most solutions are bespoke and require intense theoretical and practical modelling. The sheer range and volume of solutions grows virtually at parity with the problems they seek to solve. Nevertheless, swarm robotics promises to be an area of great expansion in the near future, with applications ranging from practical implementations such as military, industrial, municipal, and agricultural, to less tangible applications as cyber-security, internet protocols, and others.

### Foraging

Navigation, namely how to interpret sensor data and plan a route accordingly, is one of the most common issues in modern robotics. Exhibiting numerous and varied applications, the task of *foraging* requires location of and navigation to a ‘target’, and subsequently return to a ‘base’. Here, swarm systems often have an advantage: an individual agent may have

weak navigational capabilities, but if one agent has located the target, then all others can make use of the information and act accordingly. In the case of *cluttered environments*, where obstacles and objects can obstruct locomotion, sensing, and communication, the ability of a swarm to explore multiple locales simultaneously can offer advantages which even the most advanced single agent cannot attain. In this respect, swarm robotics systems, particularly those emulating biological systems such as eusocial insects, are a natural fit to solve the issue of foraging.

The primary issue facing swarm solutions to foraging is overcoming the limited capabilities of agents. Swarm robots traditionally have very limited sensing and processing capabilities, which generally precludes the implementation of advanced localisation, communication, or navigation processes. Traditional methods of navigation such as SLAM are entirely unfeasible due the advanced hardware and computational requirements; as such, more novel solutions must be found.

Many swarm-based solutions to foraging look to biologically-inspired processes to succeed, generally relying on some manner of marker trail which is subject to positive reinforcement (a process known as stigmergy), but these are often subject to hardware or environmental limitation. Alternative solutions have been proposed and developed, but are largely in their infancy, and no comprehensive comparison has been found.

This work aims to implement a series of non-stigmergic methods and assess their performance against each other in a series of environments exhibiting varying degrees of clutter. Five algorithms are tested, loosely grouped into three types: Random Walks, Encounter-Mediated, and Information-Mediated. These roughly equate to the amount of information transferred between agents. Additionally, the optimal values for some important parameters are also investigated, with brief analysis of how these vary with swarm size and environment type. Conclusions will be drawn about the most appropriate algorithm for different scenarios, and what can be gleaned from this comparative testing.

## Contributions

This work offers four main contributions to the field in its execution:

The first is the application of established algorithms to environments outside of their original design, onto cluttered environments, and analysis of the results. Of the three algorithms emulated from literature, only one was originally tested in what can be deemed cluttered environments, and then only in simple ones. Extending these implementations into more complex environments (and in one case into non-idealised robots) provided in-



sight as to the direction some foraging algorithms should take in the future when dealing with non-idealised scenarios.

The second contribution is the comparison between different algorithms using a central performance metric. While this appears to lack novelty, most algorithms have been analysed using differing environments, metrics and, in some cases, manners of implementation. Emulating these algorithms as best as possible on the same swarm and in the same environments offers a standardised comparison. Additionally, an impartial analysis of their performance has highlighted some issues which are not immediately obvious. It should be noted that this work has not sought to discredit any of the algorithms implemented here, which were chosen initially for their promise in solving a highly complex issue; but in some cases, their application outside of their initial design has highlighted flaws which would be critical in real-world applications, which (it is this author's opinion) should always be the goal when designing systems such as these.

The third contribution is that of an analysis of the effects of some parameters on swarm performance. During the process of choosing optimal parameters for values such as the update interval for communication between agents, analysis of results gave some insight as to the effects of environment on these optimal values. As this was more a by-product of the tuning process, this is not a comprehensive analysis, but could be useful in top-down design for foraging algorithms in the future, as the effects of changing some important parameters is recorded. Section 4.3 records the variation of optimal parameters with environment, as well as the proportional change in swarm performance.

The final main contribution made is the development of a new algorithm, dubbed the 'Composite' method. This algorithm combines aspects of two of the emulated methods, with an additional factor developed solely by the author in the form of developing an analysis of the clutter factor of an environment by taking the ratio of an agent's euclidean distance from its target to the distance it has actually travelled. This algorithm was shown to have strong performance in all arenas, outperforming all other algorithms in most scenarios, and rivalling the performance of the best otherwise. A more detailed explanation can be found in Section 3.2.6.

### Organisation

This rest of this work is organised as follows. Chapter 2 offers greater detail on the history and background information relevant to the work undertaken, as well as a review of relevant literature. Chapter 3 describes the experimental setup, implementation of each algorithm, and the tuning process. Chapter 4 investigates the performance of all algorithms under a variety of different conditions and provides analysis, as well as a brief investigation of the

effects on performance of optimally-tuning parameters against a standard profile. Chapter 5 offers conclusions, including the perceived weakness of this investigation and avenues of potential development in future work.

# Chapter 2

## Background

### 2.1 Swarm Theory

#### 2.1.1 Swarm Robotics

Swarm robotics involves the use of a large number of simple, similar, autonomous robots working collaboratively to perform tasks more effectively than can be achieved alone. Crucially, to differentiate from other areas of multi-agent robotics, swarm robotics relies on decentralised control where simple interactions between robots result in more complex ‘emergent behaviours’. There is a consensus amongst those who have sought to classify and formalise swarm robotics that it seeks to generate systems that are robust, flexible, and scalable [2]. Robustness is the ability to continue function despite loss of individuals; flexibility is the ability to adapt to different environments and tasks (and incidentally represents the biggest challenge of swarm robotic systems); and scalability the ability of the swarm to operate largely unchanged at various group sizes. To this end Sahin [3] lists five fundamental qualities for Swarm Robots:

- Agents must be individual, autonomous robots.
- There must be a large number of robots in operation. It should be noted that some definitions such as those given by Brambilla et al. do not include this constraint [2].
- There should be few, homogeneous groups of robots. This largely precludes the use of leaders or specialised roles within the swarm, though this is not always the case [4].
- Robots should be either incapable of executing the assigned task independently, or else performance should be dramatically improved by the use of a swarm [5].
- Robots should only have local sensing and communication abilities. This ensures decentralised control.

Navarro and Matía [6] also use Iocchi et al.’s [7] taxonomy to classify swarm robotic systems as Cooperative, Aware, Strongly (or Weakly) Coordinated, and Distributed. Here ‘aware’ implies knowledge of other robots’ existence, and distributed is a synonym for decentralised. This makes the requirements of large numbers and homogeneous robots (or groups of robots) less stringent.

Distributed sensing, task parallelisation, the ability to solve problems impossible for single-agent systems, and robustness to agent failures [6] are benefits that are largely the same as those of any multi-agent robotic system. However, swarm robotics offers the additional benefits of being able to operate completely autonomously, without any manner of central control system (allowing operation in ‘foreign’ or unseen environments) as well as offering an increased cost to benefit ratio per robot deployed.

### Swarm Behaviours

Though their flexibility is debatable, swarm robotic systems have proven to have a variety of applications. Both Navarro and Matía [6], and Brambilla et al. [2] list a number of tasks that swarms can be set to. These tasks include forming groups or patterns (aggregation), covering an area evenly (dispersion), chain formation [8,9], flocking, task allocation [10–12], collective decision-making [13], source searching, collective transport of objects [14], and collective mapping [15]. It is often the case that practical applications do not map one-to-one to these behaviours; for instance, foraging in cluttered environments, the main focus of this study, is largely an issue of collective mapping and decision making (to determine the shortest path), with chain formation also being an observed behaviour for some solutions.

#### 2.1.2 Biological inspiration

Almost every study of swarm robotics begins with reference to the biological systems which are the major source of inspiration [4,5,8,9,12,13,15–34]. While it is tempting to shuck this tradition out of sheer fatigue, it is simply not feasible when the two areas are, undeniably, intrinsically linked. From inspiration to final form, swarm robotics is often about emulating biological analogues whilst simultaneously overcoming the shortfalls of implementing these designs onto systems that have not had the benefit of several hundreds of millions of years of natural evolution and optimisation.

Biological ‘swarms’, be they ants, bees, termites, fish, birds, or else, represent observable, macroscopic case studies [35]. While the inner workings of an individual may be relatively complex and difficult to observe, the actions and behaviours of a group of animals working in conjunction with each other can be observed in practical terms, and often

in relatively short periods of time. From a mathematical point of view, it is believed that swarm action can provide an increase benefit vs. cost ratio [35] to individuals, as well as to the swarm as a whole.

Flocks of birds and schools of fish have been modelled historically for some time and in a number of different ways [32], but it is eusocial insects which represent generally the most complex swarm behaviour (aside, debatably, from humans) in biology. Ants, termites, and honeybees all demonstrate advanced social behaviour, Garnier et al. [28] cite ant armies and termite nests as examples of behaviours that belie the simplicity of the individuals involved:

‘A single insect is not able to assess a global situation, to centralize information about the state of its entire colony and then to control the tasks to be done by the other workers. There is no supervisor in these colonies. *A social insect colony is rather like a decentralized system made of autonomous units*’ [emphasis mine].

It is no coincidence that this is practically the definition of swarm robotics.

Foraging is the swarm behaviour most synonymous with the work in this paper. Two of the most famous foraging mechanisms in natural studies are those of ants and bees, which manifest in very different ways. Ant foraging is generally done over an area with proportionally high densities of agents and correspondingly more frequent interactions in the environment, while honeybee foraging operates over a much sparser region, with virtually all interactions taking place at a central nest.

## Ants

The three main foraging methods of ants are encounter-based [36], stigmergic [28], and site fidelity [5]:

Encounter-based behaviour is observed most notably in Argentine ants *Linepithema humile* and *Lasius fuliginosus*, and describes behaviour that is largely mediated by the rate of encounters between agents. Argentine ants forage in a random walk pattern, with more frequent changes to direction when they perceive a greater number of ants nearby. *Lasius fuliginosus* exhibits behaviour which is specifically designed to maximimise encounters, such as wall-following, and has been observed clustering when the density of ants in a region is low [36].

Stigmergic behaviour is the hallmark of ant foraging; it can be disputed that site fidelity, the tendency of individuals and the swarm to return to known sites based on limited individual memory is more a direct subset of stigmergy than an individual behaviour in its own right. Stigmergy is defined as ‘indirect communication mediated by modifications

of the environment’ [37] and represents a simple but incredibly powerful phenomenon. Stigmergic foraging is accomplished by the use of pheromones: as an ant searches for food, it leaves a trail detectable by other ants which are then influenced positively to follow the path of the first ant. This pheromone-mediated foraging acts as a positive reinforcement: other ants will be encouraged to follow the trail of the original pathfinder, and will subsequently lay down trails of their own, strengthening the influence of the trail, and thus encouraging more ants. This potentially runaway positive reinforcement is moderated by the negative feedback of pheromone evaporation. This acts to encourage ants to take shorter routes: longer routes will result in greater periods of time between interactions between a specific part of the environment and the ants, and thus provides a longer time for the pheromone to evaporate. As the influence of the pheromone is directly proportional to the amount present [35], this means that shorter routes will experience greater traffic over time, further reinforcing the effect.

The major caveat to this, however, is that this method is very sensitive to initial conditions. In situations where a shorter path is introduced *after* a stigmergic trail has already been established, the ants will most frequently prefer the older, more established trail, than the newer, shorter, one [38]. Work has been conducted with an aim to overcoming this shortfall [27] as ants have no conception that they are taking the ‘wrong’ path, while human-designed robots can benefit from a greater level of cognition and awareness. An additional issue raised by Shah et al. [27] is that stigmergy lacks flexibility: it is effective for foraging but cannot easily be applied to different tasks. Despite these shortcomings, stigmergy analogues remain among the most popular implementations of swarm robotics.

## Honeybees

Honeybees’ most famous means of communicating during foraging is the waggle-dance [39,40]. Upon returning to a hive after a successful foraging session, a honeybee will conduct a ‘dance’ which indicates the direction, distance, and quality of a food source to other bees. Despite being one of the more remarkable forms of communication observed in nature, there is suggestion that the waggle-dance does not operate as might be expected. A large proportion of foragers ‘recruited’ by the waggle dance do not find the indicated source [35] but instead seem to operate on their own initiative based on a behaviour akin to site fidelity. As a result, it appears that some papers emulating the waggle dance as a foraging method suffer from perhaps an incomplete understanding of the full workings involved. As Pitonakova et al. [20] posit ‘[Their] results point to the importance of studying the context in which a particular behaviour is used in a swarm, especially when nature-inspired behaviours are taken out of their biological context and are applied in an engineered robot

control algorithm.’ They particularly caution against combining multiple behaviours in modular ways, warning that new behavioural mechanisms can interact with old ones in unique (and therefore unpredictable) ways. Additionally, the fact that the waggle dance in its original form only deals with direction and distance means it is likely unsuitable for cluttered environments, as the most direct path to a food source does not necessarily encourage the shortest possible route.

### Termites

Turner [12] champions termites as an underutilised inspiration in swarm robotics but it appears that, while termites do represent systems which exhibit complex self-organisation, they are not so suited for studies on foraging. There is little literature on their foraging habits beyond pheromone use, and there is even evidence that termites’ foraging methods habitually do not find the shortest path [41]. Whether this is indicative of a true lack of observable behaviours, or simply a lack of research, is unclear. If it is the latter, it is perhaps a cruel coincidence that the lack of literature on the creature’s foraging habits can be attributed to an initial bias and subsequent positive reinforcement towards ant studies.

## 2.2 Swarm Mechanics

### 2.2.1 Emergent Behaviour

Emergent behaviour is defined by Webb [42] as ‘Interesting capability that is not explicitly programmed but emerges from interaction of sets of reactive behaviours, the robot with the environment, and multiple robots with each other’. More succinctly, Sumpter summarises that ‘Simple interactions [can] produce complex patterns’ [43] and that swarms embody the concept of a ‘whole beyond its parts’ [44]. Practical examples of emergent behaviour (sometimes called ‘self-organisation’ in literature) are too numerous to list, but notable examples include boids ‘flocking’ from three basic rules (cohesion, separation, and alignment) [32], ant cemetery formation from simple object transport choices [16], dynamic chain formation from foraging behaviours [45], and the construction of termite mounds from detection of gas concentrations [12]. In all cases, it is generally the case that the final behaviour does not directly follow from the individual rules, and cannot arise without interactions between the agents, either directly or indirectly.

Navarro and Matía [6] set the criteria for self organisation: positive feedback, negative feedback, randomness, and multiple interactions. In the case of ants foraging above, these four criteria are explicitly met: positive feedback is provided by the ants’ deposition of pheromones on trails they are following, negative feedback comes from the evaporation of

pheromones, randomness from the fact that an individual ant's choice is still only probabilistic given a choice of two paths with varying amounts of pheromone [35], and multiple interactions come from the fact that any ant can deposit or detect the pheromones. Thus, stigmergic foraging is an absolute textbook example of self-organisation or emergent behaviour.

Despite being well-defined in qualitative terms, emergent behaviour is not well-suited to quantitative analysis. Most analysis focuses on either analytical observation by the experimentalist [9] or plotting of relevant metrics over time or by number of robots, which do not represent flexible analyses and are often extremely task-specific [16]. Sperati et al. do use an analysis based on entropy [45], but it is once again based on the heading direction of robots which is most relevant to their study (dynamic chain formation) but lacks extensibility.

The most quantitative description to be found is that of Sumpter which describes emergent behaviour as exhibiting a phase transition [46], where an observed factor becomes organised, most usually in time. This observed factor does not have to necessarily be the behaviour of the group, but can be the formation of structures, as with the aforementioned termite mounds. Sumpter asserts that some emergent behaviour can be modelled via differential equations, but does not go into detail on the topic, and uses the specific example of foraging ants, which may indicate that once again, there is a heavy reliance on task-specific metrics to observe emergent behaviour and define when this phase transition has occurred.

### 2.2.2 Development Methods

Swarm robotics has suffered from many of the issues that emerging fields are commonly afflicted by; most notably, a lack of formalism and structure has meant that research methods and proposed solutions have been extremely varied in approach and implementation. While this has resulted in an excellent variety of methods to solve common problems, few make a habit of building on others, meaning that a reasonably amount of work on a researcher's part must come from personal inspiration.

When designing swarm robotic systems, the most common distinction is made between the microscopic or individual level, and the macroscopic or swarm level. Approaches to swarm robotics tend to focus on one of these scales individually, in an effort to affect the other. Whether an approach is ground-up or top-down (local-to-global or global-to-local respectively in Sumpter's parlance [46]) can dramatically affect the methods employed. It



can be easy to specify a specific behaviour for an individual to exhibit, or an overall behaviour required for the swarm. The relationship between the two can be complicated, and this is an area of ongoing research. Winfield et al. [47] note the difficulty in transitioning from micro- to macroscopic methods, and that swarm systems are inherently non-linear. The paper is largely successful but necessarily makes some simplifying assumptions, and uses a method which is again specific to the tasks the swarm is employed on. It is likely that no general solution to the issue of the relationship between microscopic and macroscopic behaviours can be found due to the massive variation in task, environment, and the highly non-linear behaviours of swarms. Sumpter seems to support this view: despite reiterating that most biologically inspired swarms share deep similarities, he recommends a pragmatic, case-by-case approach to swarm modelling [43]. This is partially supported by Hecker et al. [24], who focus more on the relationship between swarm design and real-world applications, stating that the majority of systems are designed under ideal conditions and then adapted to the real world, rather than being designed from scratch with the issues of practical application in mind.

Brambilla et al. [2] assert that at their time of writing (circa. 2013) there are ‘still no formal or precise ways to design individual level behaviors that produce the desired collective behavior’. Their work, however, does represent the most comprehensive categorisation of design approaches seen in swarm robotics as a field. They split designs into *behaviour-based* and *automatic*. Due to their natures, the two methods are usually ground-up and top-down respectively.

Behaviour-based design is the most common and most intuitive method, representing methods that are implemented and changed almost exclusively by the designer, often making use of mathematical models or biological inspiration. These include systems which model swarms as self-propelled particles where the majority of interactions are modelled by potential fields [48], but the most common example of behaviour-based design is the probabilistic finite state machine [47, 49]. These designs can become very complex, but have a reasonably intuitive basis: a set of behaviours or states are defined, with transition functions being affected by probability functions derived from the sensors. Arguably the simplest versions of these are ‘prescriptive’ robots which act in a defined manner based on their sensors, with no clearly defined behavioural states, and are purely reactive.

Automatic design is more complex but potentially very powerful, employing evolutionary algorithms [21, 24, 50] or reinforcement learning. These methods are naturally more adaptive to change than behaviour-based methods, but require a large amount of tuning and often a large number of generations or learning period to become fully viable. In situations where there is not scope for learning, or else a more prescriptive design is just as

appropriate, behaviour-based design is entirely suitable.

All methods used in this work represent behaviour-based design.

### 2.2.3 Communication

Communication is a fundamental aspect of swarm robotics; without some form of communication the concept of emergent behaviour is effectively null. Communication does not necessarily have to be direct communication between agents [7], it can be a passive sharing of information either through the environment (as in stigmergy) or from the robot itself, where information is broadcast indiscriminately, and it is up to individual agents to process and interpret that information. There are two aspects of swarm communication to consider, inter-agent communication, and the spread of information to the entire swarm, corresponding to the microscopic and macroscopic effects. As with most areas of swarm robotics, predicting the effects of changes to the system at one level and how they will affect the other is entirely nontrivial.

There is considerable variability in the methods employed for inter-agent communication, even within the same task. Methods range from direct communication between agents [9] to passive communication via LED lights (indicating current task and direction) [4] to simply registering the presence of other agents [26] or even simply ignoring other agents altogether and only interacting with stigmergic traces [25]. While, naturally, greater levels of communication offer superior cognition, swarm robots are often by necessity simpler than many other types of robot, and so there may not be practical scope for complex communication [18, 51]. Stirling and Floreano [52] investigate the effects of different combinations of sensors on the energy efficiency of a swarm: firstly robots with neither communication, nor sensing capabilities; then with low-bandwidth communication only; and finally with only sensing capabilities to detect other robots. At this stage of research, efficiency is normally a consideration secondary to performance but it is worthy of note that they found that all three methods reduced ‘energy consumption’ of the system, proving that even with no communication or greater sensing capabilities, appropriate deployment of the robots can act to increase efficiency.

Stigmergic methods often aim to bridge the shortcoming of direct communication but their implementations often lack substantial robustness. Methods either rely on some kind of evaporating physical system (marker pens [33], alcohol [25], glowing paint [22], and gas [34] have been used) or a virtual implementation [19, 29]. The problems with the former methods are numerous and serious; physical interpretations of pheromones are largely im-

practical to implement onto robotic systems - no system demonstrates capabilities which are acceptable. Virtual methods show more promise and have the benefit of unambiguous signals, but generally rely on a dedicated environment to implement.

Other, non-stigmergic methods of indirect communication have been tested; Ducatelle et al. [4] use multiple LEDs to indicate direction which is successful but relies upon a second swarm to act as surveillance, Manor and Bruckenstein et al. [53] use a bearing-only system with the ability for agents to detect a global signal and discern its direction, though the implementation appears somewhat limited. The most advanced implementation of non-direct communication is found is Kasprzok et al. [18] which uses a combination of direction and registering presence. Using only these passive communication methods, Kasprzok et al. claim to have achieved a level of foraging efficiency which appears comparable to stigmergic methods. However, it should be noted that most of these studies are constructed in environments which suit their design and therefore there is currently no effective means of performance comparison between differing communication methods.

The issue of macroscopic communication is effectively that of the ‘flow’ of communication. This is more concerned with the overall method of communication than the mechanical fineries. Haque et al. [54] make a distinction between visual and topographical methods. Topological communication is a system whereby communication is mediated by cartesian distance to the agent (similar to a K nearest neighbours implementation), visual is based on the sensory capabilities of the robot. Haque et al. found that different methods are more appropriate for specific tasks, and that visual was most appropriate for the ‘search for a goal’ task. Crucially, their tests did not include the implementation of obstacles which would logically affect the outcome. It is unclear if topological communication is affected by obstacles, as visual communication would undoubtedly be.

Pitonakova et al. [20,23] examine the macroscopic differences between the waggle-dance and stigmergic methods. The waggle-dance, while not inherently direct communication, does involve a much more direct transfer of information, and thus is more suited to a centralised method to maximise exposure to other agents; stigmergic behaviour is much more decentralised by its complete lack of necessity for active inter-agent communication. The major conclusions are that the waggle-dance method exhibits better foraging plasticity in dynamic environments, but tend to over-exploit fewer, more condensed sources. Perhaps most importantly, the studies reinforce that swarm robotics currently is not particularly proficient at fulfilling its own mandate of generating flexible systems: specific designs work well for individual tasks but often suffer greatly when that task is altered.

A brief note on the use of leaders: some studies have used leaders in implementations, where a specific agent has access to more information than others [14, 48]. This can overcome the issue of requiring simpler robots, as these advanced capabilities are often only installed on very few individuals. There is even strong evidence that ants exhibit some leader-like behaviours in some activities [36]. However, it can be debated as to whether a system employing leaders can be regarded as swarm robotics; certain it violates Sahin's definitions (the robots would not be homogeneous) and the presence of an individual with access to greater information due to enhanced capabilities can be seen as a form of centralisation, thus violating Navarro and Matía's definition as well. For the purpose of this work, leaders were not used, but the effect that enhanced individuals have on a system, particularly from a cost/benefit analysis point of view, may be interesting.

## 2.3 Foraging

Foraging is a frequently-utilised study in swarm robotics, involving the location of food 'sources', and, on occasions, the transportation of items back to a central nest. This overall task is itself very scalable and can involve multiple sources of varying quality [20]. At its most basic level, foraging represents a task of collective navigation, but can be extended to collective transport, decision making, and mapping. Foraging translates well to real-world applications: acting in foreign (i.e. unknown) and potentially dynamic environments has many analogues such as toxic waste collection, demining [55], defence, [15], civil waste collection [24], and the virtually ubiquitous application in robotics of search and rescue [1, 56]. Especially once the issue of cluttered environments is considered, foraging becomes an issue of information flow. This is an area to which swarm robotics is well suited, both from biological precedent, and a consideration of the robustness required to maintain performance in uncontrolled areas [9]. Stirling et al. [52] show that foraging behaviour can be extended to 3 dimensions; as a notable issue with flying robots is the issue of localisation, again swarms become a powerful solution when solutions can rely only on inter-agent communication and basic behaviours such as obstacle avoidance.

Swarms do, however, exhibit limitations when applied to foraging. Most notably, there is a strong dependency on the number of agents in an environment, known generally as the density. Due to the method in which swarms behave, particularly in cluttered environments, there is a necessity for a minimum density of robots to ensure there is any kind of emergent behaviour, generally known as the critical density [19, 35]. However, aside from the various issues of cost, it has been shown by Rybski et al. [57] that increasing the number

of agents results in diminishing returns after a certain point. In cases where the number of agents is very high, issues of congestion start to decrease performance dramatically [18]. Thus, there is a balance to be struck when considering the optimum density to be used in any foraging task.

### 2.3.1 Cluttered Environments

Foraging studies have traditionally taken place in otherwise empty environments, free of obstacles. There are a number of benefits to this, including potentially simpler behaviour requirements for robots and a lack of occlusion for communication. Macroscopic studies especially tend to ‘relax’ the assumption of having an environment which can interfere with communication [50], while other studies which deal with cluttered environments often have the issue of dealing with the presence of obstacles as the main focus for the task [9, 14].

From a practical point of view, cluttered environments represent several challenges: robots must be able to avoid obstacles while foraging; agents may be close in cartesian distance, but geodesically may be separated by great distances with visual communication occluded; the most direct path to the source may not be the shortest path available, this includes issues of ‘dead ends’; and the fact that communicating the concept of having to turn at junctions is not trivial, especially when robots are likely to lack any advanced form of localisation. Solutions to this issue involve either increasing frequency of communication (either by stigmergy or increasing density of robots) or else by implementing a form of memory into the robots. While the former is arguably the more traditional ‘swarm solution’ there is a limit on its effectiveness, especially when hardware and communication considerations are made.

### 2.3.2 Stigmergic Methods

Stigmergic methods of foraging are generally the most common, likely owing to their popular biological origin. According to Sumpter [35], stigmergic foraging was first analysed by Wilson [58], who also established that pheromone trails can act to follow the shortest route. Since then, multiple emulations of stigmergic foraging have been attempted.

Nouyan et al. (2008) [8] use robots as virtual markers, forming chains which can be altered over time depending on whether the chain is connected to the source. The system used only passive communication, using coloured LEDs and proximity sensors. The end result is an unbroken chain of robots between the nest and source which can be used for navigation by other robots. While this method is effective, and could theoretically cope with cluttered or even dynamic environments, it has several notable shortfalls. The first is that the system will find a local minimum, and not necessarily the shortest path, once

the source is located there is no incentive for the swarm to find any improvements, the task is treated as finished. The second issue is that of wastage: while the process may be reasonably robust to robot failures (the chain will likely re-form over time), the use of robots as stigmergic markers is massively wasteful, as a large proportion of the swarm is reduced to complete immobility. The final issue is that of congestion, using immobile robots as markers uses up space which, in cluttered environments, can sometimes be at a premium.

Mayet et al. (2010) [22] used paint which glows under UV light to simulate stigmergic traces, while this represents one of the more successful methods of actual physical emulation of stigmergy, it suffers from a requirement that the entire environment be covered in the specialised substance and therefore is unsuitable for foreign environments. Otherwise, the method is simple practical evidence that stigmergic methods are feasible to implement on robots.

Ducatelle et al. (2010) [4] use a heterogeneous swarm, with ceiling-based ‘eye-bots’ forming a dynamic chain which alters its position based on the position of the foraging ‘foot-bots’ in a form of dual-feedback. It also deals explicitly with cluttered environments. While powerful, and technically not wasteful of robots in the way [8] is, the system nevertheless appears to have some obvious issues. The first is that, while a successful chain will represent a successful navigation system, the chain will naturally tend to a local minimum, rather than finding the global shortest route. A further issue is that the system is once again largely reliant on environmental setup. The use of aerial markers necessitates a large degree of unobstructed open space above the obstacles the system is trying to avoid, which is not always available or practical. Finally, this is still technically use of robots as semi-static markers in the swarm, though these robots are not acting as additional obstacles as the robots in [8] are.

Khaliq et al. (2014) [29] use a similar method to [22], employing RFID tags. Once again, though the method is successful in what it set out to achieve its requirement of a compliant environment (one with embedded RFID tags) and thus is entirely unsuited to foreign environments. In cases where an environment can be modified to such a level that thousands of RFID tags can be implemented, it is surely expected that more precise methods than simple stochastic, stigmergic foraging can be implemented. These stigmergic systems suffer from their physical requirements making their solutions redundant.

Fujisawa et al. (2014) [25] employ robots that release trails of alcohol, emulating the pheromone trails. While these break from the compliant environment requirements of previous methods, it does introduce the additional issues of hardware to both carry, deploy, and detect the alcohol trails. Additionally, the issue of a limited quantity of material, though not crippling, is of some concern. Given that stigmergy is presented as an altern-

ative to direct communication, it seems counter-productive to employ a system which has even more hardware requirements.

Llenas et al. (2018) [19] implement ‘virtual pheromones’ in an advanced system and conceptually impressive system which nonetheless suffers from the same issue as Mayet and Khaliq: requiring a controlled environment. This method also effectively relies upon a centralised system to distribute the information to the agents, debatably violating one of the principles of swarm robotics. Additionally, this implementation (and others similar to it) all have the problem that they offer no real advancement: stigmergic foraging is a process optimised by millions of years of natural selection, it is entirely expected that emulating it almost exactly will be successful. Past a ‘proof of concept’ that it can be implemented onto real robots, these methods offer no real advantage, and can be argued to be not taking advantage of some of the capabilities of robots, most notably virtually instantaneous, unambiguous communication from a distance.

While other stigmergic studies are available, it is clear that most studies in stigmergy either suffer from requiring a specialised environment to operate in, precluding the possibility of operation in foreign environments, or require specialised and potentially unreliable hardware. The most promising work is in fact one of the earliest methods, which used the robots themselves as stigmergic markers (the appropriateness of designating this a ‘stigmergic’ method can be debated), but this had the notable issues of wasting a large proportion of the swarm as navigation markers, and lacking an ability to find a shortest path, as opposed to the first path found by the swarm (effectively a local minimum). It was determined that methods explicitly emulating stigmergy are not the most appropriate for foraging studies.

### 2.3.3 Waggle Dance

The waggle dance is a form of direct communication, and when applied in foraging, it is generally performed at the nest. While there is a strong suggestion that the waggle dance is more a method of recruitment than strictly information dissemination [35], the practical applications are much the same. Robots employing waggle-dance have a benefit that they can provide unambiguous information to others, but have the additional issue of still requiring simultaneity both spatially and temporally (stigmergy only requires spatial) as well as requiring dedicated communications hardware.

Comparatively less studies have been done on waggle-dance foraging. Bailis et al. (2010) [30] represent one of the very few studies to use solely the waggle dance method in foraging, and their study relies primarily on the issue of determining the effects of

the waggle dance on multiple, varying quality food sources, rather than as a method for navigation. Thus the waggle-dance method here is treated as more of a method of collective decision-making rather than navigation. Other studies by Pitonakova et al. (2016, 2018) [20, 23] confirm this, using completely non-cluttered environments, and comparing the waggle dance to stigmergy primarily in terms of its effects on the plasticity of the swarm (ie. its ability to adapt to dynamic environments). In [23], Pitonakova et al. also list situations in which various forms of stigmergy or waggle dance are the most appropriate, potentially aiding the design of behaviours for specific tasks.

### 2.3.4 Other non-stigmergic methods

Given that both stigmergy and the waggle dance have considerable downsides, and that neither method appears strongly appropriate for foraging in robot swarms, other methods must be considered.

#### Random Walks

Naturally, a purely random walk would not be an appropriate model for swarm robotics, lacking any form of inter-agent communication. However, advanced random walks, where some of the parameters of the random walk are mediated by the robot’s interaction, could prove useful, especially in the initial ‘exploration’ stages of foraging. Dimidov et al. (2016) [59] show that, in searching for a source, a Correlated Random Walk proves to be the most efficient method of searching. The study is most notable in that it represents one of the truly original methods of foraging control, though the authors do note that it is fundamentally a baseline study, with many potential avenues for further study and improvement. While the CRW is not likely to be the most efficient means of foraging, it is a reliable starting point, particularly if one is trying to minimise the time before other, more advanced methods can be employed after the initial discovery.

#### Site Fidelity

Site fidelity represents the ability for agents to remember the location of a source and return to it. While a powerful tool, this method relies upon a degree of dead reckoning which is not popular in swarm robotics as it puts too much emphasis on the capability of individuals rather than the swarm. However, Abbott and Engelbrecht (2014) [17] demonstrate that even a very basic implementation of site fidelity via path integration (an emulation of desert ant foraging techniques) is more capable than ‘naive’ or random foraging. In some circumstances, the site fidelity method was even shown to outperform a waggle dance



method, as the latter required a longer time period to adapt to the circumstances. None of the tests were run on cluttered environments, which is important to note.

Hecker et al. (2012) [5] run a similar comparison, but instead use a pure site fidelity method against one combined with a pheromone-inspired foraging system. Using site fidelity only, 3 robots achieved twice the foraging rate of a single robot, and 100 robots achieved a rate of 66 times greater, indicating a strongly positive impact on performance for what the authors consider to be a minimal cost to implement. The results of the pure site fidelity method vs. those of the combined method are mixed; simulator results indicate that the combined method should result in increased foraging rates, but practical implementations show decreased rates due to accumulating errors, though this may be due to an implementation which was not designed to take into account the effects of real-world implementation. Both studies show that a degree of memory implemented into robots can improve performance for, depending on hardware setups, minimal effort and implementation. However, site fidelity would be best implemented in conjunction with other methods, else there is little benefit to being part of a swarm past basic increasing numbers set to a task.

## MANET

Ducatelle et al. (2014) [9] proposed one of the most advanced forms of (quasi-)non-stigmergic foraging. The system, inspired by Mobile Ad hoc NETWORKs, uses every agent as a broadcaster for information. This information usually is simply whether an agent (not necessarily the broadcaster) has seen the source, as well as the dynamically updated time since this information was received, and distance the information has travelled either through transmission or on-board a robot. By broadcasting this information through a network of agents, and allowing receivers to choose directions based on the newest information, the system allows for swarms of robots engaged in different tasks to act as mobile stigmergic markers. It was demonstrated that the agents found the shortest route in a cluttered environment the majority of the time. However, it has two notable shortcomings. The first is that it is a capability-heavy design. The system relies heavily on direct communication, as well as accurate dead-reckoning and RSS capabilities; all of these systems increase the cost of the robots involved, and as such the cost/benefit consideration must be addressed. The second issue is actually one brought up by Llenas et al. [19] in their analysis of the work, in which they observe that the system is heavily dependent on environment size, and thus presumably agent density. Despite these issues, this method represents one of the state-of-the-art methods of dealing with foraging in cluttered environments and is one of the primary inspirations for the methods demonstrated in this work.

### Anti-Stigmergic Methods

Some studies have sought to actively avoid the use of any form of stigmergy, and mostly avoid any form of direct communication. While these studies do implement aspects of other methods seen previously, they generally use more advanced mechanisms.

Sperati et al. (2011) [31] use robots which use only passive directional signalling. The system was set up without any emphasis on cooperative behaviour, but by applying an evolutionary algorithm which rewarded the rate of foraging, it was found that dynamic chains very similar to those found in explicitly cooperative studies were formed. While the exact methodology employed may be outside the scope of this work, it is a study which offers two primary benefits: firstly that the dynamic chain behaviour is confirmed as being, evolutionarily, an optimal method of foraging; hence while the methodology is not necessarily applicable, the results are. The second consideration is that the study represents one of the very few times that emergent behaviour is quantified, in this case, in terms of entropy. Examining how relatable this may be to similar studies could be beneficial. However, this study notably makes no mention of cluttered environments, and its simplicity may mean that any meaningful information taken from it may not necessarily be applicable for methods which are not evolutionary, or else use different sensing methods.

Kasprzok et al. (2018) [18] succeeded in implementing a foraging behaviour that relied on absolutely minimal sensing, employing a variety of methods working in conjunction with each other. Though robots do broadcast if they are returning from the source and their heading, that is the only information that is actively transferred from agent to agent. The robots implement both a correlated random walk and path integration, with much of their behaviour being mediated by the rate of encounters with other agents, in conjunction with a decaying ‘memory’ which serves to weight decisions. The primary issues with this work are that its focus is spread between navigation, congestion avoidance, and recruitment strategies, and thus does not deal with any in a great deal of depth. This includes the issue of cluttered environments: while the study does use a simple, maze-like environment for part of the implementation, it is for analysis of congestion avoidance rather than achieving a shortest path. This emphasis on congestion avoidance may also be slightly redundant as Sumpter [35] maintains that an efficient flow of agents can be achieved purely from collision avoidance. The final issue is that the study lacked an implementation on real robots, which, as shown by [5] can result in different results to pure simulations. This study, along with those of Sperati et al. [31] and Ducatelle et al. [9] were the primary inspirations for new foraging techniques in this work.

### 2.3.5 Role of Errors / Randomness

In many studies, commentary is made on the role of errors in the system. While traditionally, most robotics systems seek to minimise or eliminate errors, there is evidence that errors often act as either a form of negative feedback or else add a degree of randomness to the system, two of the fundamental components of self-organisation as specified by Navarro and Matía [6]. Pitonakova et al. note that full information spread in a foraging swarm is almost always more beneficial [23] but that errors and negative feedback are crucial to maintain performance [20]. In this respect, accumulated errors often result in agents finding new sources or new paths to the same source which may be more beneficial to the swarm overall than the ‘status quo’. As such, errors as a form of negative feedback can encourage plasticity in the behaviour of the swarm, which is a key component to avoiding the initial bias which methods such as stigmergy struggle to overcome in isolation. Sumpter [35] claims that the waggle dance method in fact fails frequently, and that a large proportion of honeybees that are ‘recruited’ do not in fact travel to the communicated location with any degree of accuracy or reliability, but that this is essential to the functioning of the swarm.

On top of this, there is inherently a proportion of randomness to biological foraging methods. Sumpter shows that both stigmergic ants and honeybees can be modelled by the same equation when foraging, namely: given a choice between two paths (or two food sources in the case of bees), the probability of choosing path/site  $X$  over path/site  $Y$  is given by

$$P(X) = \frac{(x + k)^\alpha}{(x + k)^\alpha + (y + k)^\alpha} \quad (2.1)$$

where  $x$  and  $y$  are the proportional concentrations of pheromones on path  $X$  and  $Y$ , or number of bees dancing for site  $X$  and  $Y$ , and  $\alpha$  and  $k$  are constants. The values of these constants vary from species to species but Sumpter notes that for honeybees,  $\alpha$  is always modelled as unity, while, as an example, for the ant species *Lasius niger*  $k = 6, \alpha = 2$ . This model is still inherently probabilistic, indicating that the biological inspirations still rely upon a degree of randomness, which should be considered when designing swarm systems, which can rarely but occasionally be modelled as being kinetic or deterministic in their behaviours. Garnier et al. [28] support this with the assertion that ‘randomness is often crucial, because it enables the colony to discover new solutions’, encouraging the link between errors and randomness in swarm robotics systems, especially when considering swarm plasticity.

However, Hecker et al. [5, 24] note that though the negative feedback aspect of errors can be useful, in some cases error propagation results in dramatically reduced performance, supported by Bailis et al. [30]. As such, it is clear that while randomness is an essential part

of swarm robotic design, and that errors can contribute to this randomness in a positive manner, the errors must be carefully managed to avoid reducing the performance of the system they were designed to enhance.

# Chapter 3

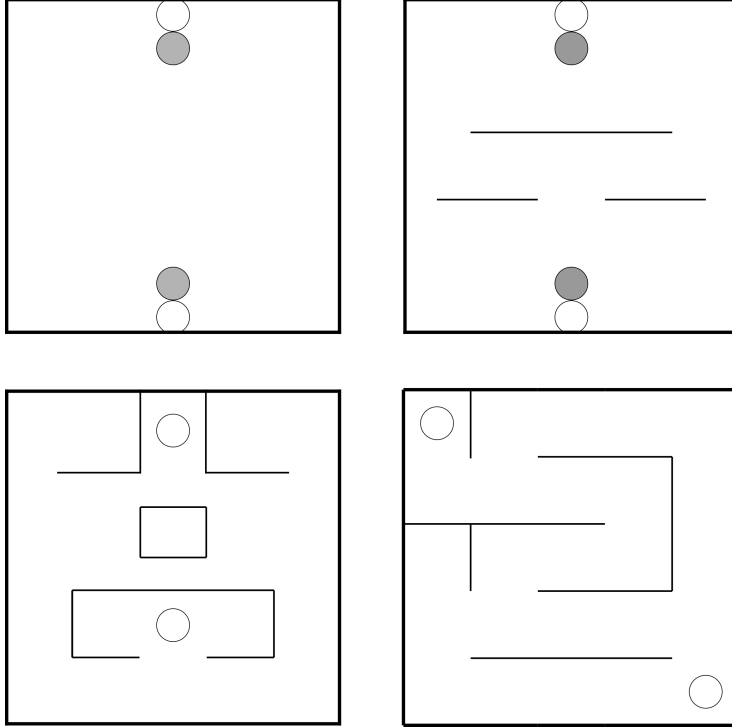
## Methods

### 3.1 Environment

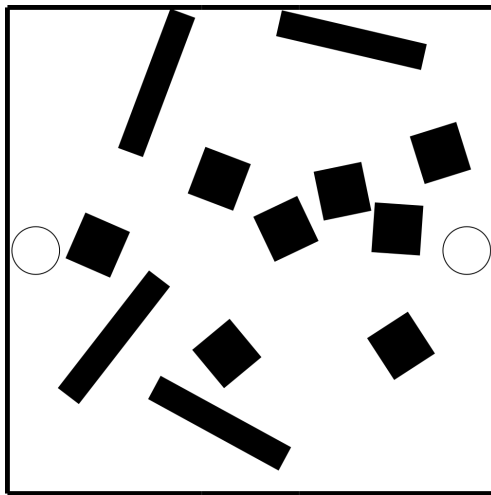
Behaviours were implemented using the ARGoS 3 simulator [60]. The simulator allows for efficient implementation of large numbers of robots in real time (or faster) with accurate physical behaviours and sensor characteristics, and is designed to be extremely modular and customisable.

The simulation used foot-bots as the agents involved in foraging. The robots possess two wheel/track hybrids (colloquially: treels) which provide locomotion up to a speed of  $30\text{ cm s}^{-1}$ , and a ring of LEDs around their circumference with an additional ‘beacon’ on top. The integrated sensors are numerous and include proximity sensors, ground sensors, a range-and-bearing system, an omnidirectional camera, distance scanner and light sensors. As mentioned, the simulator was designed to be modular and as such, not all of these were utilised.

Landmarks were originally added as coloured areas on the floor, but this restricted the ability to generate differing arenas, and so both the base and target were implemented as individual foot-bots running a simple broadcast script. Any robot which came within a radius of the robots’ broadcast was deemed to have reached the goal they represented and altered its behaviour accordingly. This system proved more flexible than the original and allowed better integration into the Ducatelle method; in a real-world application it is likely that an agent would have to dedicate itself to broadcasting information in any case. The only drawback of this method was that it was possible for the landmarks to be moved slightly by collisions with other agents.



**Figure 3.1:** Static arena designs for testing the system. Top left: Arena 1, an empty arena, without clutter ( $D_b = 1.14$ ). Top Right: Arena 2, a simple, low clutter environment ( $D_b = 1.16$ ). Bottom Left: Arena 3, a mid-to-high clutter environment ( $D_b = 1.29$ ). Bottom Right: Arena 4, a maze-like environment ( $D_b = 1.25$ ). Circles represent the location and size of the base (bottom) and target (top), shaded circles represent locations used in ‘offset nest’ tests where landmarks were deliberately placed further away from walls.



**Figure 3.2:** Example of an arena with random clutter. The same number and type of obstacles were used consistently, but their orientation and position changed between tests. In this example  $D_b = 1.481$ . The average value of 10 arenas was calculated as  $D_b = 1.474$  ( $\sigma = 0.008$ ).

As this work relied on variable environments, there was no set design for the arenas used. The only common features were that all experiments used a square arena enclosed by walls, with a single base and target. Otherwise, 5 basic arenas were used, which can be seen in Figure 3.1 and 3.2. To obtain a quantitative measure for the amount of clutter present in an arena, the box-counting fractal dimension  $D_b$  was used. Fractal dimensions generally represent a measure of complexity in a system, defined by Karperien [61] as the change in detail with change in scale.  $D_b$  specifically is ‘useful for objectively quantifying complexity in digital images’ and represents the relationship between the number and size of boxes obtained from a box-counting algorithm. The values quoted in Figure 3.1 and 3.2 represent mean  $D_b$  values generated using the Fractalac plugin for the ImageJ tool, and can be used to estimate the amount of clutter observed in an arena [56]. As an interim measure, until more detailed models are developed, Fractal Dimension serves as a quantitative measure of how cluttered an environment is, with the downside that it can only be used in 2-d and requires prior knowledge of the arena’s construction.

Additionally, two sets of initial conditions were also considered: arenas could either exhibit the base and target at the edges of the arena, or slightly offset, in order to examine the effect of not having the base near edges which could potentially bias towards a wall-following behaviour. Similarly, the initial distribution of robots was either determined to be clustered around the base, or else uniformly spread out throughout the arena. Where the agents were uniformly spread, they were initialised so that half were initially searching for the target, and half for the base. Some literature implementations only exhibited this initial setup, so its effect on performance was analysed. It is important to note that behaviours were not tuned for these tests, they were implemented to observe the effects on pre-tuned swarms.

The number of agents used in a simulation was also altered. The size of a swarm can affect performance, both positively through increased agent density affecting interaction rates, and also negatively through increased congestion. 8, 24, 40, 64, and 80 agents were used in conjunction with the conditions above. Preliminary testing showed that this range demonstrated the improvement on performance of the swarm through to the diminishing returns at higher levels.

All tests ran for a length of 2000 seconds, at a rate of 10 ticks per second.

## 3.2 Behavioural Models

### 3.2.1 Foraging

In this work two points of interest, termed *landmarks*, are placed in an arena. One is designated the *nest* or *base*, the other the *food* or *target*. The landmark the agent is currently searching for is termed its *goal*. This work uses only one target which is functionally identical to the base, aside from identification by the robots. The underlying aim of a foraging system is to maximise throughput from the base to the target and back, known as the number of *transits*, where each transit represents an agent discovering its goal. Additional concerns such as resource depletion or target selection where multiple targets are used can also be considered but are not here: the metric for success is simply the rate at which robots transit between the locations. To this end, the main issue facing the swarm is that of navigation: knowledge of where the target is, and the shortest route to it, as well as the reverse issue of navigation back to the base. The issue of congestion can also arise at higher agent densities (or more cluttered environments) but is not explicitly dealt with here.

### 3.2.2 Correlated Random Walk

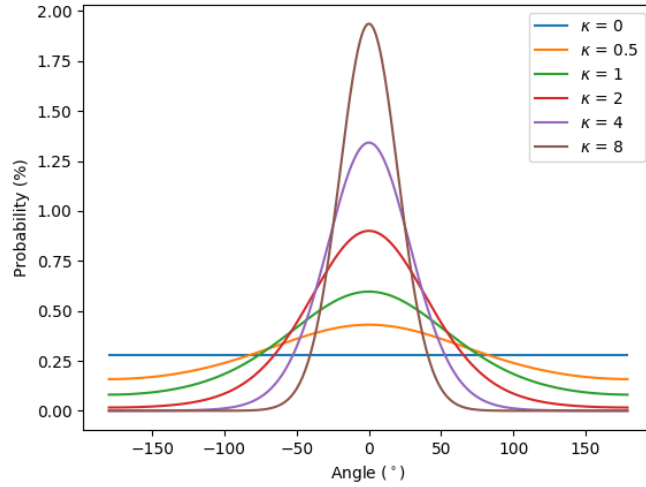
A random walk is characterised by a series of straight-line movements, called ‘steps’, with each step ending in a ‘tumble’ which changes the agent’s direction. Both the step length and the tumble direction are subject to some degree of randomness. A Correlated Random Walk (CRW) has a bias in the current direction of movement which acts to cause the agent to continue in a set direction more so than in a ‘true’ random walk. As demonstrated by Dimidov et al. [59] a correlated random walk is the best method for initial acquisition rates in swarms. As such, a CRW was generally used as a basic behaviour before other information was gained, but was also used in isolation for full tests as a baseline to compare against other, more advanced methods.

#### Step Size

In the CRW used by the agents, the step size was drawn from a Gaussian distribution of initial mean 2 m and a standard deviation which was consistently  $\frac{1}{3}$  of the mean. The mean was increased by a set value at the end of each step until the target was found for the first time. This increase over time was implemented in order to combat the possibility of a fixed step size being unsuitable for foreign environments, while the use of a standard deviation proportional to the mean allows for the possibility of smaller values still occurring.



**Figure 3.3:** von Mises distributions with varying values of  $\kappa$  with  $\mu$  set to 0. At low values of  $\kappa$ , the distribution becomes approximately uniform, at high values, there is a strong preference for values close to  $\mu$ .



### Heading Direction

To generate the new angle an agent will face after a completed step, a CRW generally uses a peaked, symmetrical probability distribution with a mean of  $0^\circ$ . A more ‘correlated’ walk is a distribution with a stronger peak, and correspondingly one with less deviation from the current path. Possible distributions for this include a Gaussian distribution [5], a wrapped Cauchy distribution [59], and a von Mises distribution [18]. Heading distributions in this work use a von Mises distribution, given by

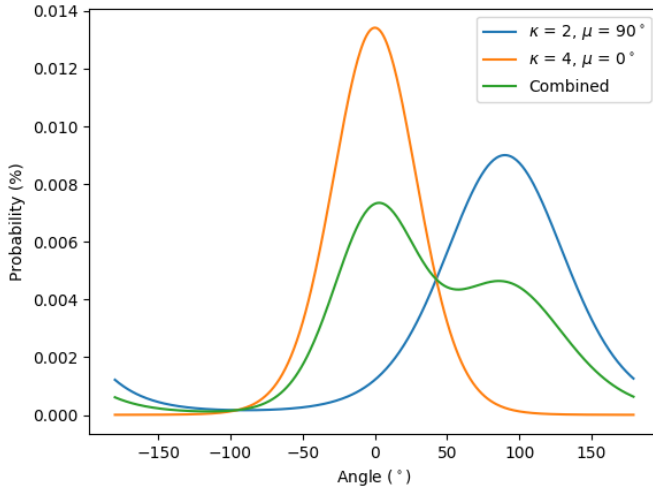
$$f(\theta) = \frac{1}{2\pi I_0(\kappa)} e^{\kappa(\cos(\theta-\mu))} \quad (3.1)$$

where  $\theta$  is an angle from  $-180^\circ$  to  $180^\circ$ ,  $\mu$  is the mean heading change (set to  $0^\circ$  for a CRW),  $I_0$  is the modified Bessel function of the first kind of order 0 given by

$$I_0(\kappa) = \sum_{n=0}^{\infty} \frac{\left(\frac{1}{4}\kappa^2\right)^n}{(n!)^2} \quad (3.2)$$

and  $\kappa$  is an adjustable parameter where  $\frac{1}{\kappa}$  is an analogue for the distribution’s standard deviation, shown in Figure 3.3. A notable difference between the von Mises distribution and a Gaussian is that at a  $\kappa$  value of 0, the distribution becomes uniform.

The von Mises distribution was chosen as it is well recorded as being useful in modelling foraging behaviour in animals [18], and it allows for simple calculation and combination of multiple radial distributions, which was necessary for the implementation of a Biased Random Walk, seen in Section 3.2.3.



**Figure 3.4:** Example combined von Mises distributions, the probabilities for each angle are summed, then normalised to generate the probability distribution function.

### Obstacle Avoidance

The CRW also implemented a basic obstacle avoidance system: agents turned away from an observed obstruction in front of them, without discerning between agents and walls. Agents then continued roughly perpendicular to the direction from which the obstruction was observed, acting as a rudimentary wall following technique. This behaviour, inherited by later methods except Kasprzok, acted to increase the rate of interaction between agents in line with biological inspiration [36]. Theoretically, this wall-following behaviour could have a detrimental effect on the exploration of large empty zones, but as obstacles did not act to reset the current step, it was possible at any point that the robot would turn away from the wall and continue.

#### 3.2.3 Site Fidelity (Biased Random Walk)

Site Fidelity uses the same basic behaviour as the Correlated Random Walk, with the additional behaviour that once the base or target is discovered, its location is recorded using a simple range and bearing system, mediated by path integration achieved by the agent's odometry systems, imitating that exhibited by ants [5, 17]. From this 'landmark bearing' value, a second von Mises distribution was generated using a new value  $\kappa_b$ , with the mean value being the calculated direction of the agent's goal. This distribution was then added to the original CRW distribution using  $\kappa_c$  and the resulting distribution normalised, shown in Figure 3.4.

The introduction of this second distribution acts to add a bias in the calculated direction of the goal each time the agents selects a new direction. The amount of bias can be controlled by the parameter  $k_b$ , which was held constant for the site fidelity tests.

### 3.2.4 Ducatelle Method (Information-mediated behaviour)

The Ducatelle method (otherwise ‘Ducatelle’) was inspired by Ducatelle et al. [9] in their work on Foraging in Cluttered Environments. The implementation can be found in their paper and has been emulated as closely as possible here; however, it is possible that some details may have been overlooked or misunderstood, or that assumptions have had to be made in the face of missing information, so this representation should be treated as an interpretation, rather than a perfect recreation. The same can be said of the Kasprzak method in Section 3.2.5.

#### Initial Discovery

In the original implementation, a robot discovering the target would then become a broadcaster, presumably staying stationary in the process, but this was simplified by having the landmarks themselves (base or target) be robots capable of broadcasting from the beginning. The landmarks constantly broadcast their identity, as well as a sequence number, beginning at 0 and increasing by 1 each time a message is broadcast.

The method begins with all agents conducting a correlated random walk as they search for their goal. Once a landmark is discovered, an agent updates its own internal information with the calculated distance to the landmark, obtained from the range and bearing module, as well as the sequence number received. Each agent can possess information about both landmarks simultaneously. As it moves, the robot continuously increases its current distance value  $d_L$  in accordance with its odometry, while the sequence number  $t_L$  remains unchanged from when it last was in communication with the landmark. The agent then periodically broadcasts that it has ‘seen’ the landmark, as well as the updated distance value and sequence number. This message is transmitted via the onboard range and bearing system, which allows 10 bytes of data to be sent with every broadcast.

#### Processing Data

Each roaming robot possesses its own table for the current best values of each landmark’s distance and sequence number, though these do not contain any data until initialised by either locating the landmark or receiving data from another robot. Upon receiving a broadcast from robot  $A$ , a searching robot  $S$  will process the data, adding the range of transmission to the calculated distance, and then comparing it to its current onboard data about the landmark. If either the sequence number is greater ( $t_A > t_S$ ) or the sequence number is the same and the distance is less ( $t_A = t_S$ ,  $d_A < d_S$ ) then the new data is considered better, and the robot updates its own information with the received data. The robot’s internal information is then broadcast periodically as normal, without the altera-

tion of the sequence number. Additionally, the agent attempts to move directly towards the location (estimated by the range and bearing information provided) of the agent which provided the best data, with a small angular offset added to avoid direct collisions.

This process, reminiscent of a Manet system, theoretically allows the information about the shortest route to ‘flood’ through the swarm in the case of densely connected systems, or be carried from location to location and communicated in the case of sparse swarms. The use of sequence numbers allows the relative recency of data to be established and prioritised, while the distance accomplishes a similar task, with the additional benefit of acting as an upper bound on the distance from the source.

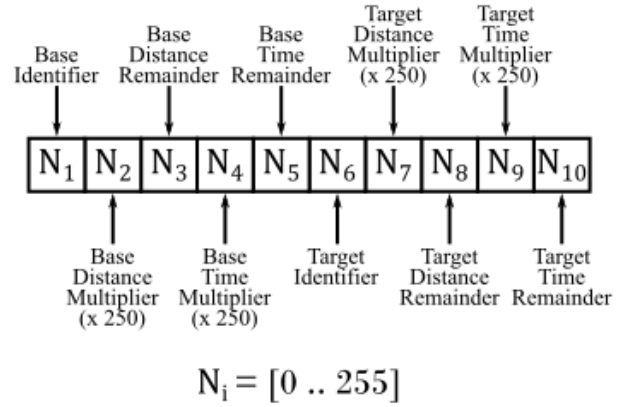
### Transmission Formatting

The transmission system allowed for 10 bytes of data to be sent every broadcast, required by the foot-bots’ range and bearing system to be formatted as 10 individual integers from 0-255. This presented a minor challenge as many of the sequence numbers and distances involved were well in excess of this limitation and the system did not automatically convert larger numbers into multiple bytes; attempting to use any value outside of these conditions simply resulted in a numerical error. The large number of values that were required to be sent concurrently (target identification, target distance, target sequence number, base identification, base distance, base sequence number) also effectively precluded representing larger numbers through simple addition. These restrictions necessitated the use of a formatting process, which can be seen in Figure 3.5. The sequence number and distance values were effectively converted into base 250 which allowed for a range of values to be communicated at a high level of precision, while conforming to the 10-byte bandwidth limit. This allowed for simultaneous broadcast of the entirety of each agent’s internal information without sacrificing information quality. Base 250 was used instead of base 256 as this made the data more easily human-readable. The loss of the additional 6 bits was deemed an acceptable trade-off for speed and clarity when testing and debugging during implementation.

### 3.2.5 Kasprzok Method (Encounter-mediated behaviour)

The Kasprzok method (otherwise ‘Kasprzok’) was developed by Kasprzok et al. [18] and, unlike Ducatelle, does not explicitly seek to deal with cluttered environments. Unlike the Ducatelle method which mediates behaviour based on information communicated between agents in the swarm, the Kasprzok method utilises agent interactions as the primary means of altering behaviour. An interaction is classified as agents touching each other, however

**Figure 3.5:** Broadcast data from each agent using the Ducatelle Method. Data was required to be formatted before transmission due to the limited bandwidth and data type. The first value in each pair represents the integer multiples of 250 in the value to be transmitted, the second represents the integer value of such a division, e.g. 560 would be transmitted as  $\begin{bmatrix} 2 \\ 60 \end{bmatrix}$ . Sequence numbers are labelled ‘time’ purely for visual presentation.



as the foot-bots lacked bumpers, this was implemented as robots coming within 35 cm of each other. The only information exchanged during an interaction are the agents’ current tasks (whether they are ‘carrying food’ or not) and their current heading. The time of the interaction and direction from which it took place are also recorded. Note that this method uses many parameters which were tuned by the original author; effort has been made to maintain this tuning where possible.

### Potential for Passive Communication

This system is a strong candidate for the use of a visual means of communication. The interaction-based encounter is simple enough to be mediated by proximity sensors or bumpers, but the transfer of more complex information is not possible with such simplistic mechanisms. A passive sensing system using coloured lights was originally designed in which the foot-bot’s beacon acted as an indicator of its task, while the radial LEDs were set to fixed colours depending on their orientation (red for the right side of the robot, white for the front, etc.). This allowed interacting robots to determine each other’s task and orientation based on which colours could be seen. Unfortunately, due to issues with occlusion errors in the simulator, this system was discontinued, but exists as a potential means of developing the algorithm without use of direct communication. In actual implementation, the same range and bearing module as in Ducatelle was used, allowing for unambiguous communication.

### Concentration Parameter

In the Kasprzok method, agents operate on a random walk of set time step (as opposed to the distance step used elsewhere) where the angle chosen is governed by a biased random walk represented by the sum of two von Mises distributions as in the Site Fidelity method. The key difference in the Kasprzok method is that the value  $\kappa$  is dynamic, known as the

‘concentration’ parameter. The concentration parameter is calculated via the equation

$$\kappa(t) = \kappa_{min} + \sum_{i=1}^n A e^{-T(t-t_i)} \quad (3.3)$$

where  $\kappa_{min}$ ,  $A$ , and  $T$  are all parameters,  $t$  is the current time, and  $t_i$  is the time at which interaction  $i$  occurred.  $\kappa$  becomes larger with more numerous, more recent interactions, and as a result an agent with a large number of recent interactions will travel in a straighter line than one that has not recorded interactions for some time, due to  $\kappa$ ’s effect on the von Mises distribution. This effect decays over time, and after 10s interactions are deemed to have decayed beyond any meaningful impact and are ‘forgotten’.

As in Site Fidelity, once both the base and target have been located, the agents keep track of their location via path integration. These values are used to generate a new von Mises distribution which is then added to the first and normalised; the resulting distribution is used to choose the next direction. In the case that the agent is returning to base, the original CRW  $\kappa_c$  parameter is held at the value  $\kappa_{min}$ , while the bias distribution  $\kappa_b$  is adjusted in accordance with eq. 3.3, while the inverse is true for foraging: the CRW  $\kappa_c$  is affected as normal, while the target distribution  $\kappa_b$  is unchanged.

### Adjusting Speed

Unlike other methods, where the speed of the agents was treated as constant whilst moving in a straight line, Kasprzok uses  $\kappa$  as a mediator on the speed of an agent. The speed is given by

$$v(t) = v + \sum_i^n I e^{-\lambda_I(t-t_i)} - \sum_i^n R e^{-\lambda_R(t-t_i)} \quad (3.4)$$

where  $v$  is the default speed of the robot (set to  $10 \text{ cm s}^{-1}$ ), and  $I, \lambda_I, R, \lambda_R$  are all tunable parameters. The effect of this equation is that interactions with other agents encourage a rapid decrease in speed, followed by an increase. The overall effect of these two systems in conjunction is to disperse robots over time: more congested areas result in faster robots following more linear paths, resulting in a diffusion over time.

### Collision/Congestion Avoidance

While the method does not explicitly code collision avoidance, there is a congestion and wall avoidance process. In the case of congestion, if the value of  $\kappa$  exceeds a pre-defined limit (‘quorum’) then the agent calculates the average direction of its encounters in the

last 4 s, then generates a sector of either side of this direction, and reduces the probability of choosing those directions to 0. Similarly, when a wall is detected, the agent creates a 180° sector in that direction, persisting for 2.5 s, causing it to travel parallel to or away from the wall for that amount of time.

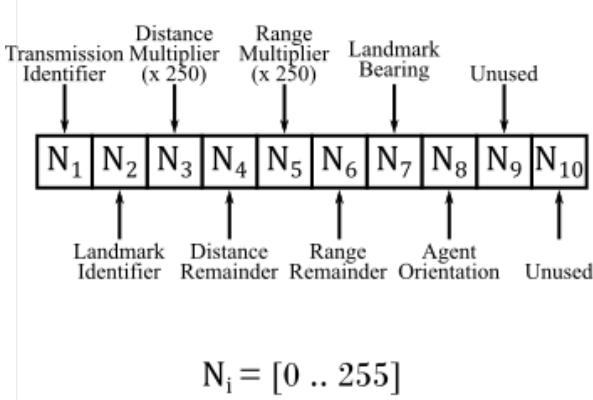
### Adjustments to Original Paper

It should be noted that while all efforts were made to implement this method as described in the original paper, the interval between which agents update their direction was originally set to 0.1 s. This was found to be completely unsuitable for the implementation as agents could rarely complete the required turn before starting forward. The original method was presumably implemented on idealised robots with an instantaneous turning ability and instantaneous acceleration. As the foot-bots used in this work did not possess such capabilities, a longer interval was chosen to better approximate this physical reality. This value was tuned manually.

The overall effect of the Kasprzok method is to prevent agents from clumping together via encouragement of both speed and a continuous direction of travel when the rate of interaction is high. As far as foraging techniques go, the system acts effectively as a biased random walk, with the additional effect that agents will aim to travel in the opposite direction to other agents which they observe are returning from their goal. The congestion effects are not directly beneficial to foraging, but can act to assist in cluttered environments where congestion is likely to be an issue.

### 3.2.6 Composite Method

The Composite method (otherwise ‘Composite’) is a largely novel method utilising features of both Kasprzok and Ducatelle in its implementation in order to overcome their shortcomings. The method is information-based, as in Ducatelle, and requires the range-and-bearing system to be fully implemented, but utilises the parameter-altering characteristics of the Kasprzok method. The primary changes are the inclusion of an agent-density clause to the Ducatelle decision-making process, only broadcasting information about the landmark the agent is coming from, and the addition of a range-to-distance ratio which affects the relative strength of the bias parameter against the correlation parameter of the random walk.



**Figure 3.6:** Broadcast data from each agent using the Composite Method. Multiplier formatting was implemented as in Ducatelle, landmark information was based on the landmark the agent is coming from (not its goal). Transmission identifier acts to distinguish agents from landmarks. The final two bytes are currently unused but could have applications through further information transfer, such as identifying agents for more advanced density analysis.

### Initial Broadcast

Upon initialisation, agents perform a correlated random walk as in all methods before. Once their goal is located, the agents begin broadcasting data. In contrast to the Ducatelle method, agents only broadcast information about the landmark they are coming from, i.e. information about the target if returning to base, and vice-versa. The information transmitted includes the distance from the landmark (as in Ducatelle, where distance is increased through travel and transmission), the range from the landmark (euclidean distance), the bearing to the landmark, and the agent's current heading (or 'desired' heading if engaged in a turning or avoidance behaviour). The format can be seen in Figure 3.6. The difference between the use of range and distance here should be especially noted, to avoid confusion going forward.

### Processing Data

Each agent transmits this data regularly, processing the information received from other agents, as in Ducatelle. The information is acted on thusly: if the searching agent  $S$  has *never* located its goal and the encountered agent  $A$  is transmitting information about the goal,  $S$  updates its own information with a range and bearing estimate to the goal which will be overwritten once it encounters the goal for the first time. It also updates its own table with  $A$ 's recorded distance to the goal, which is also done if  $S$  does have an estimate of the goal's range and bearing but  $A$  provides a lower value than its current distance estimate.

### Acting on Received Data

$S$  will then alter its behaviour accordingly only if  $A$  is transmitting information about  $S$ 's goal (landmark location estimates can be updated from any agent, regardless of current task): if  $A$  is the only robot  $S$  is in contact with,  $S$  will adjust its own heading based on



that of  $A$  according to the equation

$$h_S = (h_A + \pi) + \frac{\pi}{4} \sin(h_A - b_A) \quad (3.5)$$

where  $h_S$  is the resultant desired heading of agent  $S$ ,  $h_A$  is the transmitted heading of agent  $A$ , and  $b_A$  is the relative bearing of agent  $A$  to agent  $S$ , given natively by the range-and-bearing module. The effect of this equation is that agent  $S$  will adjust its heading to become antiparallel to  $A$ , as it is assumed travelling in the opposite direction to agents coming from a goal will, on average, lead towards that goal. The second part of the equation is to account for the potential effects of  $A$ 's location relative to  $S$ . In reality, agent  $S$  only turns strictly antiparallel with  $A$  when the heading and bearing vectors are aligned, i.e., when  $A$  is travelling either directly towards or away from  $S$ . The introduction of a slight bias towards the direction of  $A$ , proportional to the difference in the headings, allows  $S$  to align more with the path that  $A$  has taken, this is potentially most important in cluttered environments where  $A$  may be emerging from a narrow 'pathway'.

In the case where there are multiple robots broadcasting to  $S$ , it will act as in Ducatelle, moving towards the agent with the lowest calculated distance from the landmark. Sequence numbers are not used in this method. The motivation behind this threshold was that the Ducatelle method's direct navigation was observed to behave significantly better at high densities, while at low densities it was often worse than a random walk (full results shown in Section 4.2). In contrast, low densities provide optimal conditions for the antiparallel navigation which allows agents to navigate more effectively using sporadic data. The number of robots required to affect this change in behaviours is currently set at 1 but this could be adjusted or developed into a more complex measure of local density.

### Range/Distance Ratio

Additionally, it is considered that the ratio of the landmark range to the distance may be indicative of the degree of clutter in an environment. While the landmark range value is representative of the agent's euclidean displacement from the landmark, the landmark distance represents the distance the information has actually travelled. As such, the ratio of these values can tell us something about the nature of the path travelled at the time of broadcast: distance cannot be lower than range, and therefore a Range-to-Distance (R/D) ratio of unity represents a straight line with no obstacles. As the R/D ratio decreases (asymptotically towards 0), this represents a more circuitous route taken by the information; this can represent either a less linear path due to more frequent collisions between

agents or tumbling, which is hard to account for, or it can represent the agent being forced to deal with obstacles more frequently. In situations where the encounter rate between agents is low, the behaviour of the swarm is largely mediated by the biased random walk component of the algorithm, and is therefore affected by the relationship between the correlation and bias components. Where the R/D ratio is close to unity, and thus the path to the goal is clear, the bias component should take priority and the agent should prioritise travelling directly towards the goal. In contrast, as the ratio decreases, the correlation component should begin to dominate, prioritising acting on information from other agents and behaviours such as wall-following.

In practice, the R/D ratio (designated  $\rho$ ) was constrained to the range of  $(0, 1]$ . Every time the agent checked for new information, it calculated a new correlation and bias parameters by

$$\kappa_c = \frac{\kappa_{c0}}{\rho} \quad \kappa_b = \kappa_{b0}\rho. \quad (3.6)$$

Where  $\kappa_c, \kappa_b$  are the calculated values at each check, and  $\kappa_{c0}, \kappa_{b0}$  are set parameters.

### 3.2.7 Tuning

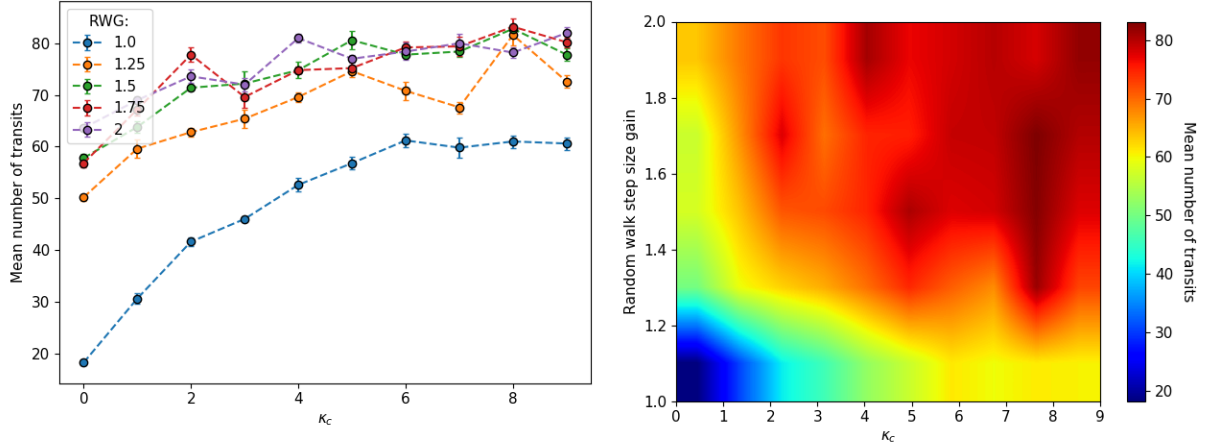
In service to brevity, the full results of the tuning process are not detailed here. Tuning trialled each parameter at a range of appropriate values on swarm sizes of 8, 24, and 64, using arenas 1, 2, and 3. The parameters required to tune were:

- Random walk step size gain (RWG) - a parameter used to counter the fact that foreign environments will necessarily be of different sizes. This parameter governed the rate at which an agent's random walk step was increased during the initial exploration phase. Too low and agents would not explore fast enough, too high and they would not explore thoroughly enough. This value was present in the CRW, Site Fidelity, Ducatelle, and Composite Methods. Kasprzok used a fixed time step set by the original author.
- $\kappa_c$  - the correlation parameter. This controls the degree of correlation in the random walk, its tendency to a straight path. This is one of the most important parameters in any behaviour as all methods rely on a correlated random walk in the initial search for the target, before any meaningful data can be shared. Once again, Kasprzok used a previously-tuned value, while in this case the Composite method used a separately-tuned value owing to its importance in the method's implementation.
- $\kappa_b$  - the bias parameter. Acting similarly to the correlation parameter, this value

controls the strength at which the bias from site fidelity acts on the random walk. Again, Kasprzok uses an existing value, and Composite was tuned separately. CRW also does not make use of this parameter.

- Kasprzok update interval - the existing implementation of the Kasprzok method calls for an update to each agent's behaviours every 0.1 s. This is impractical on real robots, so the optimal update rate was investigated.
- Ducatelle update interval - agents in the Ducatelle method broadcast their information on a regular basis, simultaneously checking received information and updating their behaviour if necessary. This rate of broadcast was shown to affect the overall behaviour of the swarm: low values lead to disorganised behaviour due to a constant flow of information resulting in rapidly changing behaviours and issues with agents following targets indefinitely, higher values mean the system simply becomes a random walk, completely nullifying the benefits of the information shared.
- Composite update interval - similar to Ducatelle, the rate of information flow of the Composite method affected the overall behaviour.
- Composite  $\kappa$  - though identical in nature to those used in the CRW, Site Fidelity, and Ducatelle method, the  $\kappa$  values in the Composite method were directly affected by the method itself via the range/distance ratio and therefore required individual tuning. Though these are labelled as  $\kappa_c$  and  $\kappa_b$  hereafter, they represent the values of  $\kappa_{c0}$  and  $\kappa_{b0}$  in Equation 3.6.

Where appropriate, values were 'inherited' by algorithms that built on previous frameworks, such as the value of the correlation parameter  $\kappa_c$  being tuned for the CRW, then inherited by Site Fidelity and Ducatelle. Independence of variables was examined early on, using the random walk step size gain (RWG) and correlation parameter  $\kappa_c$  parameters was tested. Figure 3.7 shows the results.



(a) Mean transit totals of varying combinations of RWG and  $\kappa_c$ .

(b) Heatmap showing the same data

**Figure 3.7:** Independence testing for the RWG and  $\kappa_c$  parameters. Values were tested on a 24 agent swarm using arena 2 and show the total transits of each test. Error bars represent the standard error on each value.

It cannot be said that the data shows that RWG and  $\kappa_c$  are explicitly independent, seen most easily from the heatmap, which indicates differing optimal values of  $\kappa_c$  for different values of RWG. However, each dimension follows roughly the same profile, and the left-hand plot shows that the variation is not tremendously large compared to the standard error or magnitude of the values involved. As such, the assumption of independence was upheld given the data demonstrates a mostly independent nature, against the scope of the issue of assuming otherwise, and the effort it would take to rectify given the huge number of variables involved. A full investigation of the optimal parameters has been identified as an area of improvement for future work.

The results of the tuning process gave the values found in Table 3.1 to be used in the final tests as a ‘standard profile’. This table only includes the major values that were introduced by the implementation here, and does not concern parameters inherent to the two emulated methods. All other parameters can be found in the code provided at <https://github.com/Dan-Mead/swarm-foraging>.

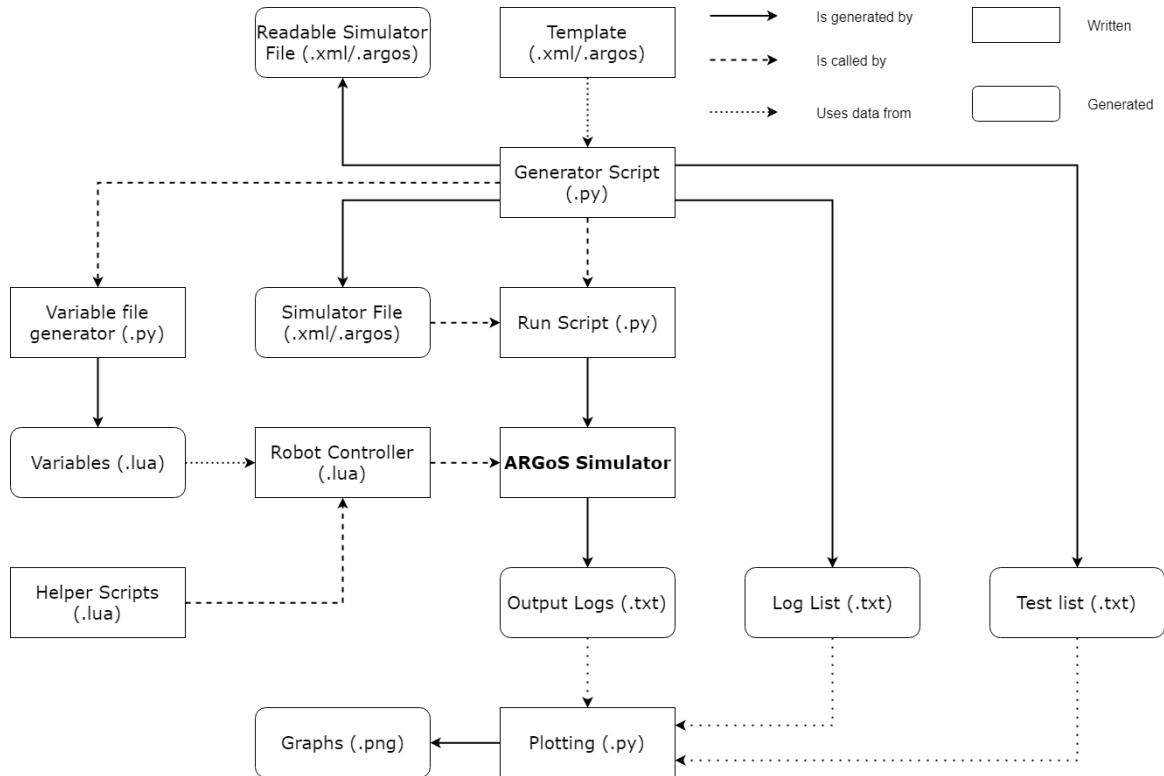
### 3.2.8 System Architecture

For the reader’s interest, the overall design of the system is provided in Figure 3.8. The process was designed to be end-to-end, but required some management when designing new experiments. On a personal note, the author is aware that many of these systems were not the optimal manner for achieving their outcome (python scripts are not traditionally

**Table 3.1:** Optimal parameter values obtained through tuning.

Parameter	Optimal Value	Notes
Random Walk Step Size Gain	1.5	Value of 1.0 used after Arenas 1 and 2, in line with testing results.
Correlation Parameter ( $\kappa_c$ )	5	
Bias Parameter ( $\kappa_b$ )	5	Identical value is purely coincidental.
Kasprzok Update Interval (s)	30	
Ducatelle Update Interval (s)	30	Identical value is purely coincidental.
Composite Correlation ( $\kappa_c$ )	5	
Composite Bias ( $\kappa_b$ )	10	
Composite Update Interval (s)	1	

used to run files through a terminal) but given the requirement of customising data inputs, flexibility in purpose, and the user’s familiarity with these systems, they were the ones that were used. This diagram is also included to potentially assist anyone attempting to replicate this work, and thus navigate the author’s occasionally convoluted system design.



**Figure 3.8:** Architecture of a Swarm: the end-to-end process of an experiment in this work. Some objects such as ‘Helper Scripts’ includes over 10 individual scripts, which were called as appropriate for different algorithms.

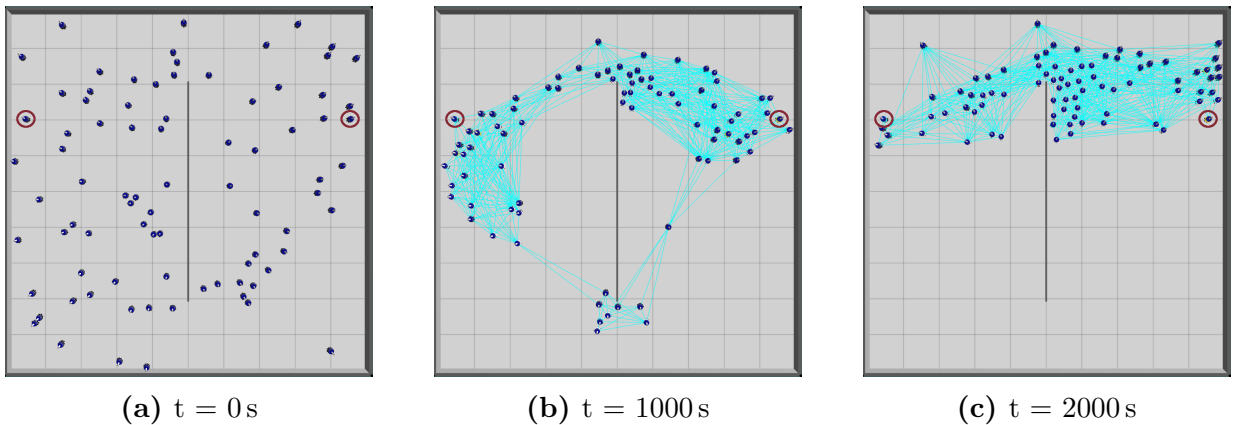


# Chapter 4

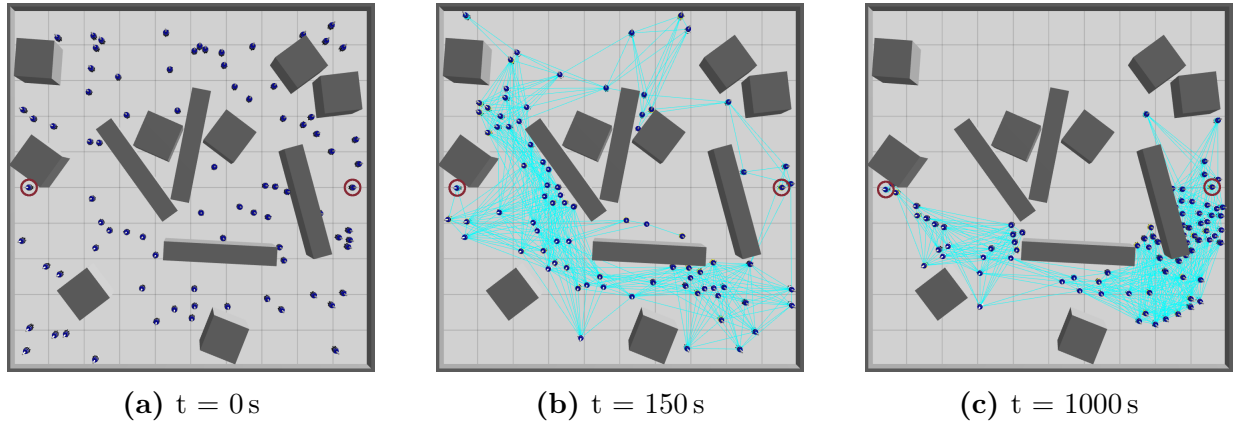
## Results

### 4.1 Analytical Results

On inspection, all algorithms behaved roughly as expected. Random walks and Kasprzok provided no interesting visual feedback; however, observations of the Ducatelle method provided clear visual evidence of the method working as intended. Figure 4.1 and 4.2 show the progression of the algorithm as the agents find the optimal route. Note particularly the issues with congestion around corners, and the fact that in the lower clutter environment, at  $t = 1000$  s some agents were still taking the longer of the two available paths. As can be seen, congestion was often an issue for this method: with no provisions for congestion avoidance, all Ducatelle agents tended towards the same, optimal path. As their basic pathfinding capabilities were limited, this led to large amounts of congestion, especially in more cluttered environments which were more likely to form ‘bottlenecks’.



**Figure 4.1:** Screenshots of the Ducatelle method in progress. The landmarks are circled, blue lines represent communications via the range-and-bearing system, corresponding to a limited-range line-of-sight system. In this setup, there was a clear shortest path between the base and target, which all agents found by the end of the simulation, despite initialising in random positions.

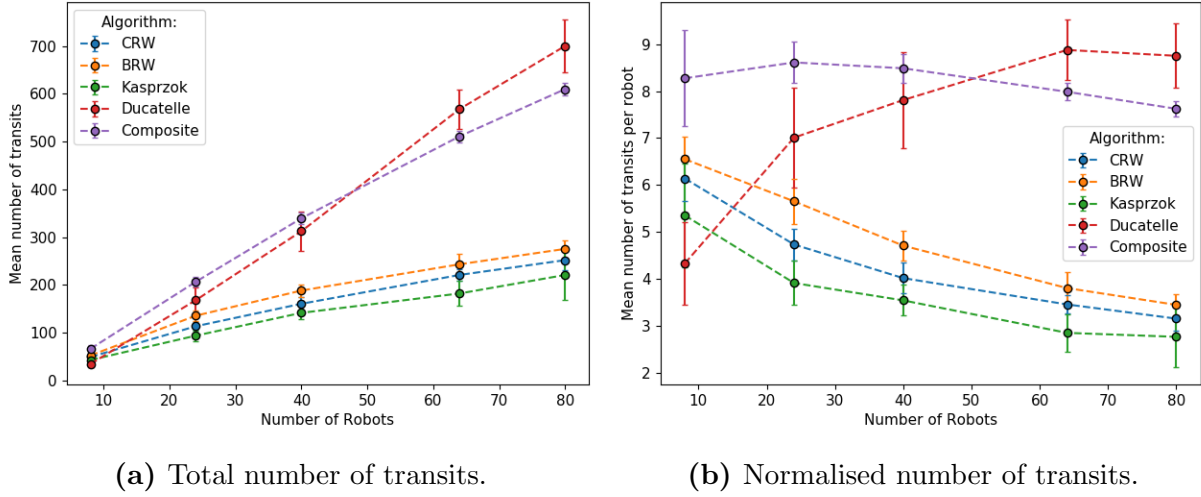


**Figure 4.2:** Similar to before, screenshots show behaviour in a higher-clutter environment. The random initialisation lead to agents exploring many non-optimal paths; over time, the swarm found the shortest route. However, the presence of heavy clutter lead to increased congestion in narrow areas.

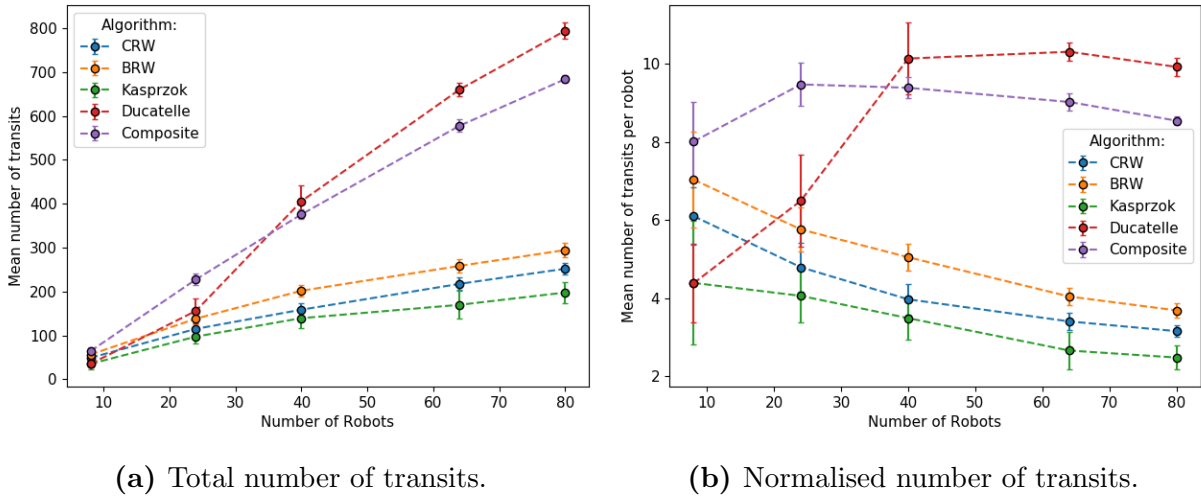
## 4.2 Arena Comparisons

All algorithms, tuned as in Section 3.2.7, were tested on the arenas detailed in Section 3.1. A total of 10 tests were run on each algorithm; in the case of the randomised arena, values were seeded to ensure the same arena was used for all algorithms. The standard deviation of each test set is shown by error bars. In all cases, the graphs show both the ‘raw’ number of total transits recorded by the agents, as well as the values normalised for the number of robots in the swarm. As all arenas were of the same size, this can be correlated with agent density. This offers an indication of how efficient individual agents were in foraging: a larger swarm will naturally exhibit more total transits than a smaller one in absolute numbers, but cooperation should ensure that larger swarms forage proportionally more efficiently, while congestion will have adverse effects. Observing the effect on individual efficiency may provide additional insight to the balance between congestion and cooperation for each algorithm.

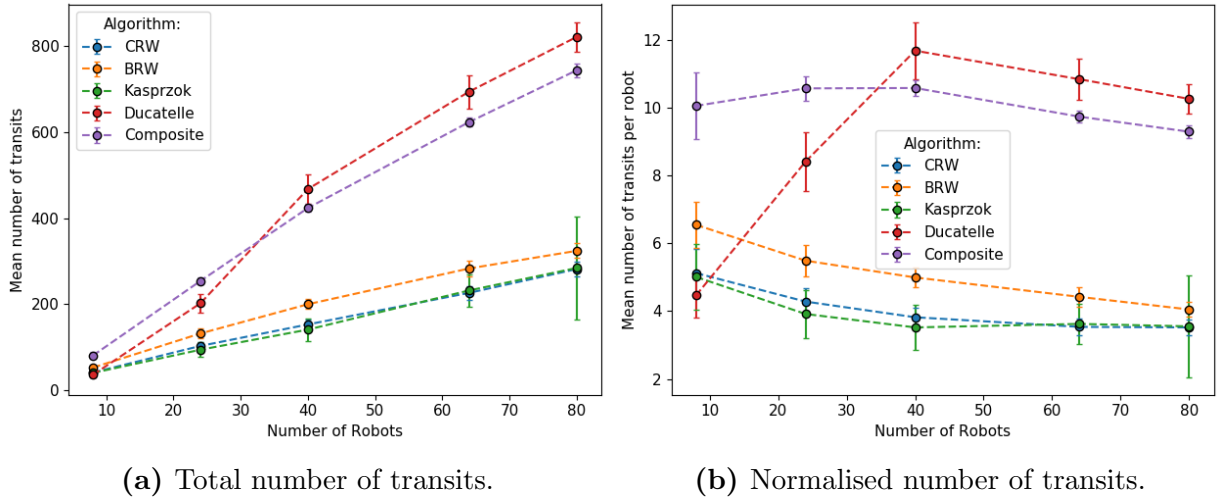


4.2.1 Arena 1 (Open Arena,  $D_b = 1.14$ )

**Figure 4.3:** Results from trials on Arena 1 with a nest-based initial distribution and the nest being edge-adjacent. The left hand graph displays the total number of transits for each swarm size, the right-hand graph shows the same results normalised for swarm size, showing relative performance per robot.



**Figure 4.4:** Results from trials on Arena 1 with a uniform initial distribution.



**Figure 4.5:** Results from trials on Arena 1 with the landmarks being offset from the edge.

Several observations are immediately evident: The Site Fidelity method (labelled Biased Random Walk in Figure 4.3-4.12) offers slightly better performance than the Correlated Random Walk. The additional bias would have no downside in an open environment, so this is expected. The normalised totals for the random walks decrease with larger swarms; as the random walks offer no inter-agent cooperation, with each robot acting completely independently, a larger number can only contribute to greater congestion, lowering performance.

What is surprising is that the Kasprzok method offers such poor performance, generally failing to achieve the rates of the basic CRW, particularly at high swarm sizes. At lower densities the Kasprzok method does briefly out-perform the Ducatelle, but this is primarily as a result of the latter's particularly poor performance in these areas. The Kasprzok's strongest performance was when the landmarks are positioned away from the walls. This was initially thought to be due to a decrease in performance by the CRW, on account of its wall-following tendencies no longer offering an advantage, as the landmarks are now out of range of any wall. However, inspection of their relative performances shows this is not the case: it is indeed due to an increase in performance by the Kasprzok method. An alternative explanation of this is that by moving the landmarks away from the wall, the congestion surrounding them has decreased, allowing the Kasprzok's congestion avoidance to relax. As many of the Kasprzok's original implementations feature arenas with the nest in the centre, this may offer insight as to why its performance is so poor compared to its theoretical value.

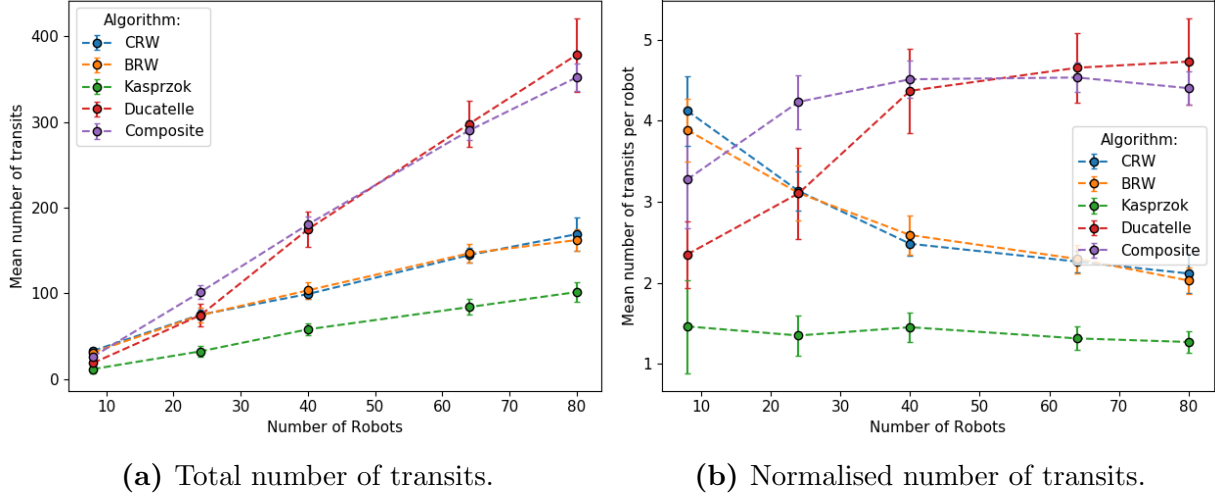
The most interesting comparison is that of the Ducatelle with the Composite method. The general trend is that the Composite method is superior for smaller swarms (8-24),

comparable for medium ones (40), and inferior for larger swarms, though the Ducatelle method is more regularly superior for the use of offset landmark locations. What is most notable is that while the Composite method is slightly inferior at high densities, it is vastly superior at low ones, due to the Ducatelle’s very poor performance, generally being the worst of all algorithms used. Given that the Composite method was specifically designed to address the weakness of Ducatelle at low densities, this result is expected, but the heightened performance is still encouraging: even at low densities, the per-agent performance of the Composite method is comparable to that of Ducatelle at high densities. Both the Composite and Ducatelle methods exhibit a slight decrease in per-agent performance with larger swarms, but generally not on the same scale as other algorithms. Composite especially exhibits the most consistent performance throughout, indicating its performance increases linearly with the number of agents used. Thus the cooperation between agents is enough to overcome the congestion, which is not observed in any other algorithms.

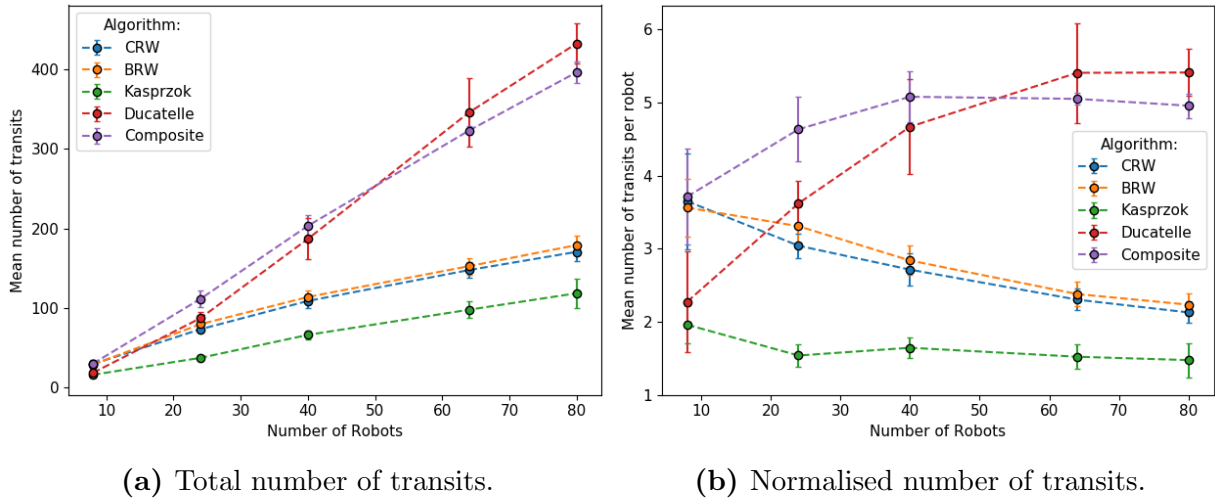
It is interesting to note that all algorithms aside from those based on a random walk performed better when the base was offset from the wall. The reason for this is unclear. It is possible that, as a result of this shift, the distances between the landmarks was reduced slightly, which could account for an increased throughput, or the issue may be that the extra directions that the landmarks could be approached from reduced congestion overall.

No meaningful difference in results can be seen from the uniform initial distribution compared to the nest-based initial distribution, aside from a slightly higher performance by the information-mediated methods. This is attributed to a lower initial time for any agent to discover the landmarks, leading to a more rapid dissemination of information through the swarm, followed by a more regular inflow of new information. The importance of information throughput is discussed further in the results for Arena 3.

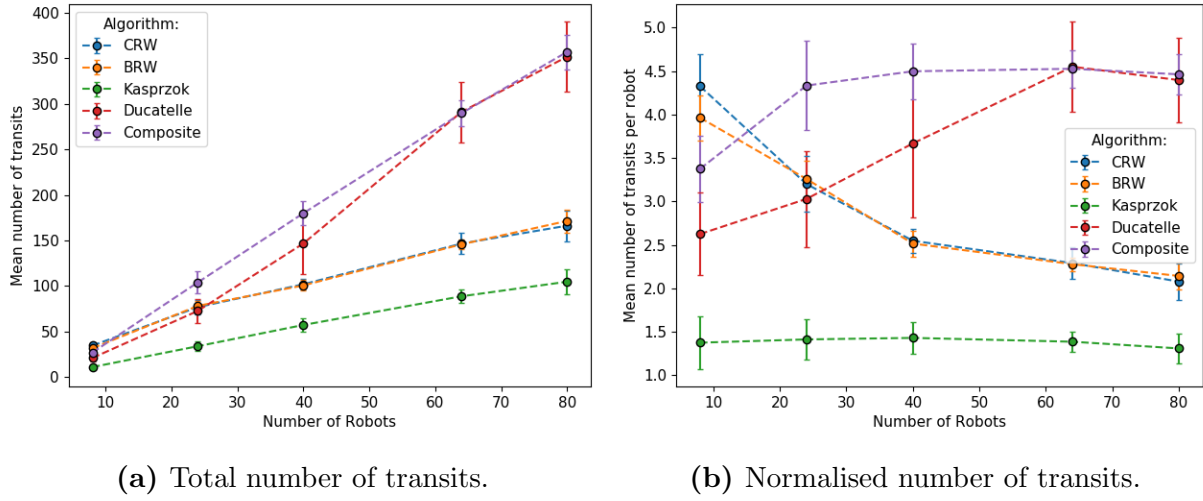
Finally, it is interesting to note that for the random walk and encounter-mediated methods, the lowest swarm sizes offered the greatest return per agent, for the Ducatelle method it was either the 40- or 64- agent swarm, and for the Composite method, it was consistently the 24-agent swarm. It can be assumed that the higher a method’s optimal return, the better its cooperation/congestion relationship. As such it is interesting that Ducatelle returns the highest, as, of the non-random walk methods, it is the only one that has absolutely no emphasis on congestion avoidance. Despite this, it is the strongest performer at higher swarm sizes, indicating that the cooperation benefits of the Ducatelle method are significantly stronger than the congestion penalties.

4.2.2 Arena 2 (Low Clutter,  $D_b = 1.16$ )

**Figure 4.6:** Results from trials on Arena 2 with a nest-based initial distribution and the landmarks being edge-adjacent.



**Figure 4.7:** Results from trials on Arena 2 with a uniform initial distribution.



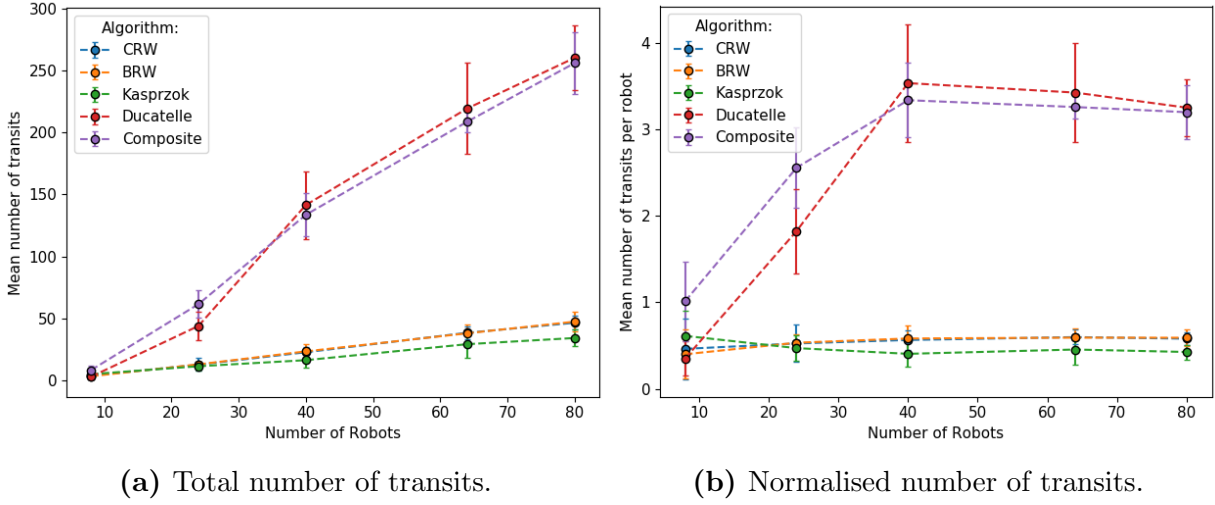
**Figure 4.8:** Results from trials on Arena 2 with the landmarks being offset from the edge.

As expected, all algorithms have reduced performance once clutter is introduced. Notably, here the Random Walk methods offer superior performance to all algorithms at lower swarm sizes. Additionally, the Correlated Random Walk is occasionally superior to the Biased. While counter-intuitive initially, this makes sense when one considers that a bias is not necessarily beneficial when clutter is considered - the most direct path is not always the shortest route to the landmark, or even one that will reach it. For this low level of clutter, the effect is less pronounced but it is expected to see this effect increasing with clutter levels.

The Kasprzok method continues to under-perform. The exact reason for this is still unclear, but it may be that the idealised design simply did not translate well to practical robots, or that the lack of focus on foraging efficiency may be showing through at this point. However, it is observed that the algorithm is largely consistent in its return per agent, indicating the congestion aspect of the method is generally successful.

The relation between the Ducatelle and Composite methods is reasonably similar to before, though the Composite method is now often comparable or even superior at higher swarm sizes compared to the empty arena. This comes at the cost of poorer performance for smaller swarms compared to previously, though this levels out at increased swarm sizes, showing a return to the linear performance. Once again, the uniform distribution offered slightly increased performance for these information-mediated behaviours, but otherwise exhibited little difference to the nest-based initial distribution.

### 4.2.3 Arena 3 (High Clutter, $D_b = 1.29$ )



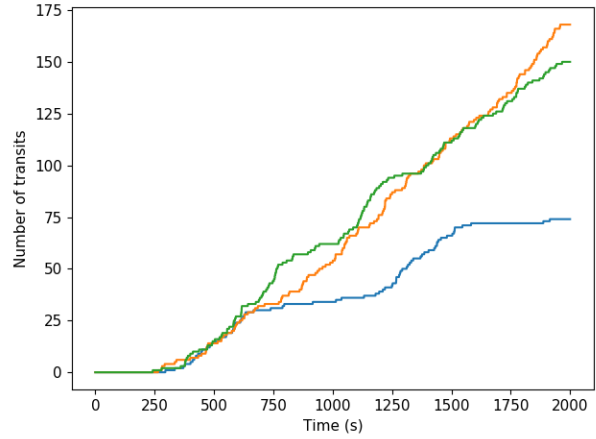
**Figure 4.9:** Results from trials on Arena 3.

Here, the performance of the Random Walk method begins to seriously suffer. Their return rates, both absolute and normalised, are extremely low; the mean transits per robot displaying an average value of less than unity indicates that not all robots even discover the base in the allowed time. The Kasprzok method shows performance on parity with the Random Walk methods.

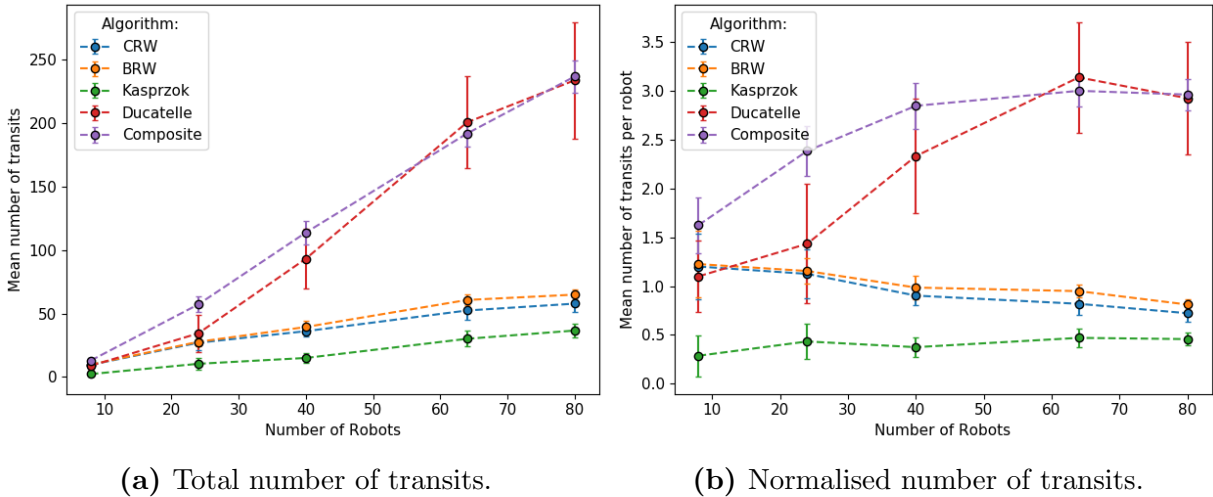
The information-mediated methods, however, retain a strong performance, though weaker than in the less cluttered arenas. The major point to note is the extremely large ranges observed in the Ducatelle method. The cause of this was identified by inspection of individual trials; the effect is shown clearly in Figure 4.10 where only one trial exhibits a constant increase in transits, the other two display, to varying degrees, periods of mostly inactivity, followed by sharp increases in the number of returning agents. Visual inspection of the behaviour during these periods shows that in some circumstances, agents can form clusters with no link to their target landmark. The effect of this is that the same information is shared between all robots, and the agents will subsequently attempt to move towards the agent in the cluster with the lowest calculated distance, which varies as all the agents turn and move. This behaviour results in an isolated cluster moving randomly around. Until new information reaches this cluster in the form of a new agent with information about the base (and even then this is usually not enough to break the behaviour, more often the robot just joins the cluster) this cluster will tend to stay cohesive and move around until it randomly finds the base. At this point there is an observably sudden increase in the number of transits, followed by a protracted period of few returns as the behaviour continues.

This highlights the importance of a constant throughput of information in the Ducatelle method: if all agents possess the same information, the behaviour can quickly become a hindrance, as the clustering prevents effective exploration. The Composite method did not generally exhibit this behaviour, making it more reliable overall. It should, however, be recognised that given the Ducatelle method exhibits improved performance on average even in the face of this destructive behaviour, its performance when this effect was not present must have been considerably superior to the Composite method.

**Figure 4.10:** Examples of cumulative returns of the Ducatelle method in Arena 3, with a 40-robot swarm. Each line represents the cumulative returns of a different individual test. Visual inspection attributed the observed jumps in returns to isolated clusters moving randomly due to a lack of new information, occasionally locating a landmark as a group.



#### 4.2.4 Arena 4 (Maze, $D_b = 1.25$ )



**Figure 4.11:** Results from trials on Arena 4.

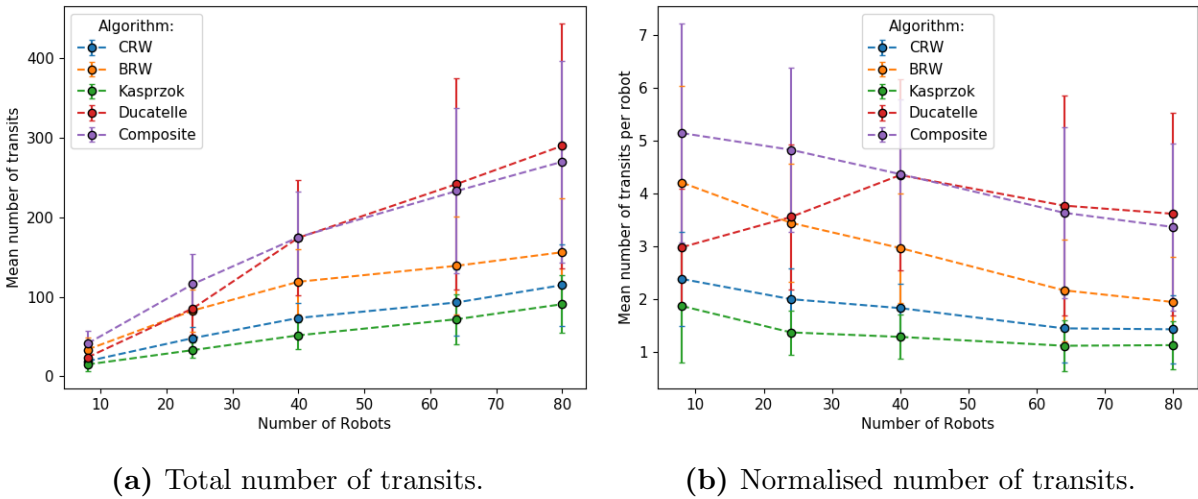
It should be noted that this arena exhibits a slightly lower clutter level than Arena 3. With this in mind, the behaviour of the Random Walks is consistent with previous tests, though their performance is notably higher than in the previous arena. The BRW does seem to

exhibit slightly superior performance despite the several ‘dead ends’ the maze arena exhibits which would logically hinder the BRW.

The Kasprzok method displays its poorest performance of all tests, with fewer than half the robots finding the goal even once. It is likely that the corridor-based arena causes the method’s congestion avoidance to affect performance in an extremely negative manner, preventing any kind of effective flow of agents.

This arena exhibits the lowest absolute performance of the information-mediated methods. The Composite is superior for longer in this case, with favourable returns even at high swarm sizes. Coupled with a generally lower variance, this indicates a generally more reliable performance than the Ducatelle, indicating this is the first time where the Composite method can be considered consistently the superior method of foraging when all swarm sizes are considered. The Composite method also shows that the returns per-agent stagnate after a swarm of around 40 agents for this arena, indicating a high impact of congestion on performance.

#### 4.2.5 Random Arenas ( $D_b = 1.473$ ( $\sigma = 0.008$ ))



**Figure 4.12:** Results from trials on random arenas (Fig. 3.2).

As would be expected, the random arenas exhibit extremely large variations. Due to the nature of the randomly-constructed arenas, it is possible that some generations created arenas where there was no available route between the base and the target. This was unavoidable but should not have affected the comparisons between behaviours, as all algorithms would have been subjected to the same arenas. With this in mind, it is interesting



to observe that the highest clutter factor by far does not result in the lowest performance. In fact, in the case of Kasprzok, its performance is better than either Arena 3 or 4 or some trials of 2. Here, the Biased Random walk is consistently superior to the Correlated, being better than or comparable to Ducatelle at lower swarm sizes.

All behaviours aside from Ducatelle exhibit exclusively diminishing returns with increasing swarm sizes, Ducatelle retains its trend of improved performance with increased swarm size early on, though it decreases commensurately with the Composite method later. Most notably, the variation and mean values of both the Ducatelle and Composite values are comparable, showing that, once again, the Composite method can be considered a superior method overall.

#### 4.2.6 Overall Observations

The first point to address is the performance of the Kasprzok method. The method, almost without fail, provided the poorest performance in all arenas. It showed to be consistently worse than, and only occasionally on parity with, random walks. It can be observed, however, that its performance is the most consistent on a per-agent basis, indicating that it overcomes the detrimental effects of congestion that a simple random walk does not. However, whether this contributes to its decreased overall performance is unclear. It may be that the trade-off for overcoming the issue of congestion is to make the system worse overall, nullifying the benefit completely. Additionally, while every effort was made to emulate the method as closely as possible (aside from the required changes mentioned in 3.2.5), it is possible that some missing detail lead to a sub-optimal implementation. The large number of parameters also indicates a reliance on careful tuning, which does not promote a flexible system. A final option is that the system was simply not well-suited for the task of foraging in cluttered environments, particularly with the implementation on real robots. As the system was not explicitly designed with this in mind (unlike Ducatelle) the lack of performance can potentially be explained as such.

Random walks were generally employed as a means for comparison of other algorithms, having no inter-agent interaction of any sort. The two main points of interest are that, at low swarm sizes, a random walk was often the most optimal solution, outperforming even the information-mediated behaviours. At higher swarm sizes, the issues with congestion became evident, but the fact that the Ducatelle and Composite methods were so vastly superior meant this was no longer an issue.

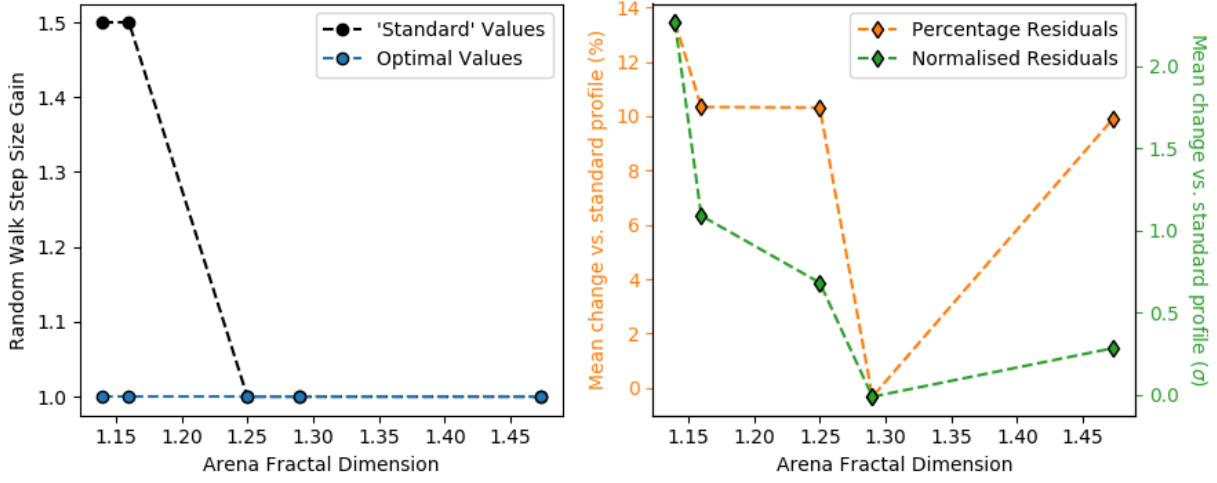
The exact interplay between the Site Fidelity/Biased Random Walk method and the

Correlated walk varies from test to test, with no discernible correlation. The BRW was more frequently superior, but this was not always the case. It should be noted that where the CRW was superior, the difference between the methods was slight, especially compared to the exhibited standard deviations, likely indicating functionally identical performance that was subject to natural variations. In contrast, in cases such as Figure 3.2, where the Biased random walk was superior, it was often vastly so. It appears that despite the issues that clutter can present when using a Biased Random Walk, it is still, overall the superior option.

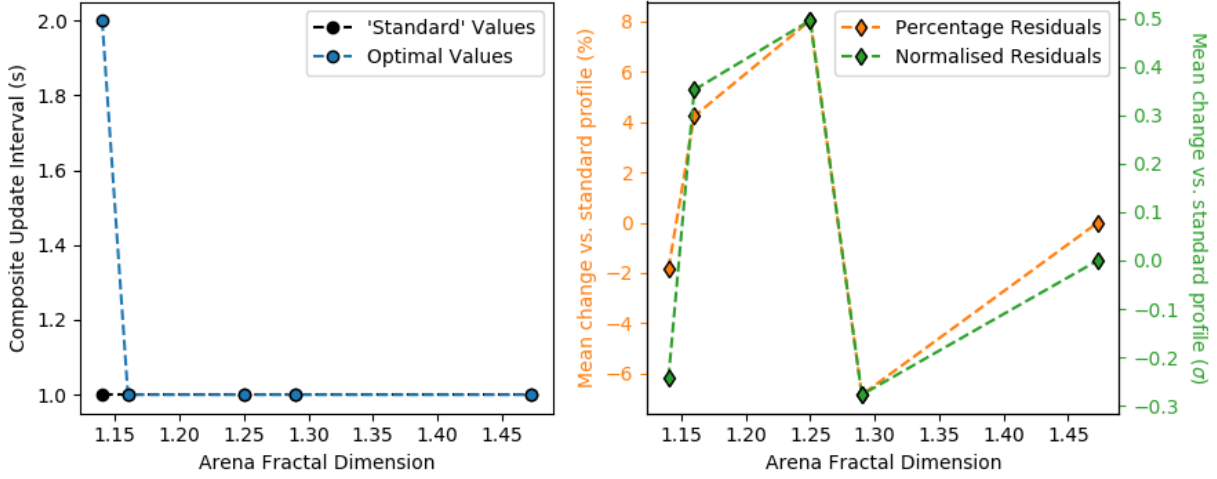
It appears that the Composite method was successful in its initial goal to improve on the Ducatelle method’s performance at lower agent densities. Except at 8-agent swarms, either the Ducatelle or the Composite were always the superior method of foraging. At larger sizes, the Ducatelle method tended to dominate, but not to the same magnitude as Composite did at smaller ones. Additionally, both methods generally exhibited diminishing returns for larger swarms, though the rates varied. The variation in the Composite method was generally lower than that of the Ducatelle method, potentially highlighting it as a more reliable technique in the face of foreign environments. In arenas with higher levels of clutter, the Composite method exhibited comparable (within displayed variations) or superior performance at all swarm sizes.

### 4.3 Changes in optimal parameters

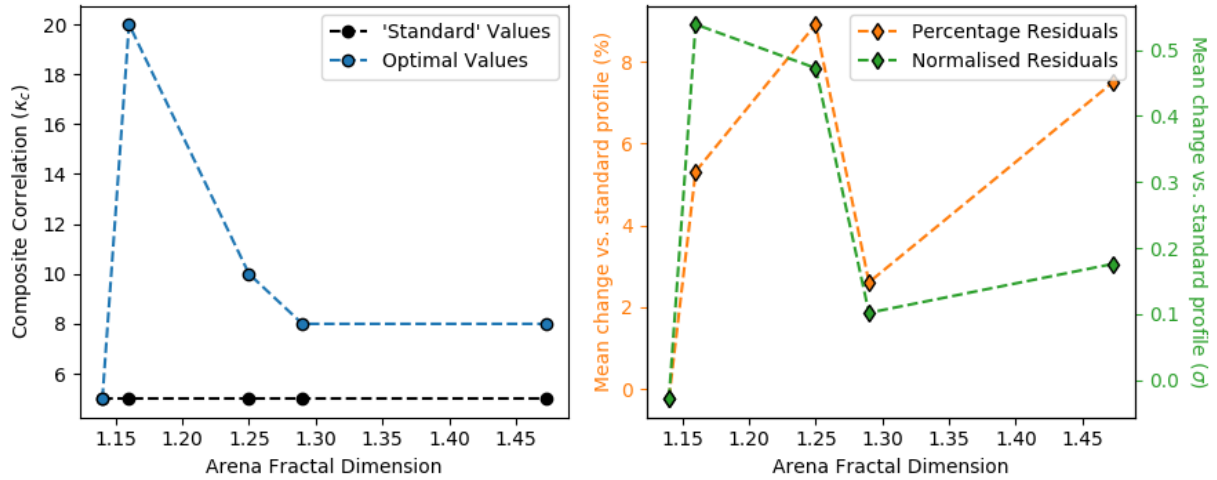
As briefly mentioned in Section 3.2.7, algorithms required tuning for optimal performance. As well as the issue with very high dimensionality due to the number of parameters involved, there was also the issue of tuning parameters for a specific arena. While efforts were made to tune on a variety of arenas, in some cases the optimal value for a parameter was considerably different between environments. This was most notable for the Random Walk Step Size Gain parameter: returns showed a positive correlation with RWG for low-clutter environments, but a negative one for high-clutter ones. While the exact relationship is likely more complicated than a simple flip in correlation (involving the choice for the initial step length, based on observations), it highlights the issue. RWG was the only parameter which was given different values between environments, as other parameters exhibited less strongly correlated values. To examine the effect that different environments could have, the optimal value for each of the parameters of the Composite method was examined per environment. All tests used a 24-agent swarm, with 10 repeats. Due to the amount of data required, a limited number of values were used for each parameter, so the true optimal value may have been between those displayed in Figure 4.13.



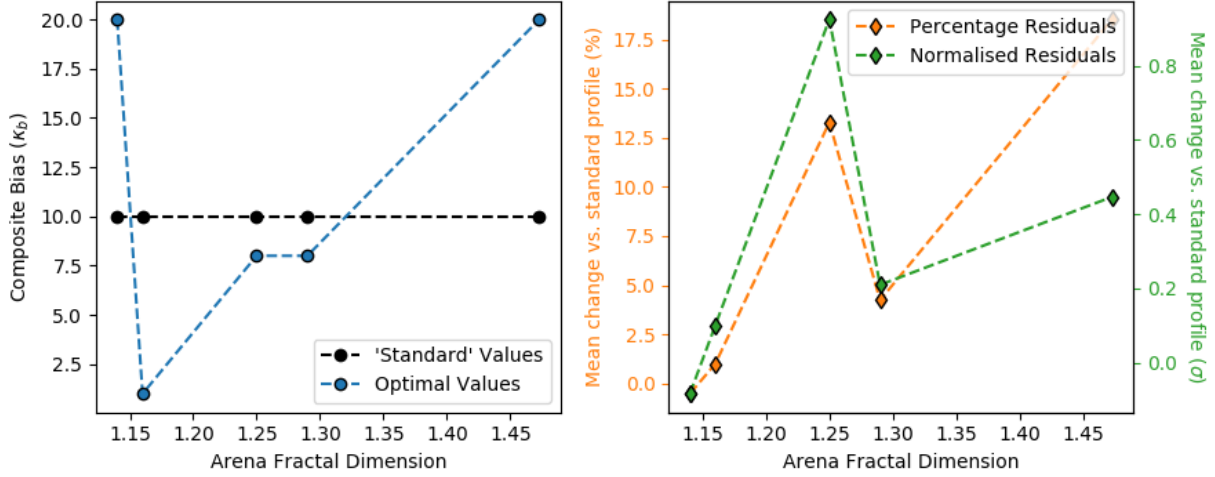
(a) Optimal values of the Random Walk Step Length Gain (RWG) parameter (left) with residuals (right), both normalised and percentage.



(b) Optimal values of the Composite Update Interval parameter (left) with residuals (right), both normalised and percentage.



(c) Optimal values of the Composite Method's Correlation Parameter (left) with residuals (right), both normalised and percentage.



(d) Optimal values of the Composite Method's Bias Parameter (left) with residuals (right), both normalised and percentage.

**Figure 4.13:** Demonstrations of how the optimal parameters can vary between environments. Residuals represent the difference between tests using the corresponding optimal value, and the value taken from the 'standard profile' in Table 3.1. Percentage residuals show the difference as a percentage of the 'standard' results, normalised residuals show the difference as a multiple of the standard deviation given by Equation 4.2. The shown values are only for the Composite method with a 24-agent swarm.

**Table 4.1:** Average residuals from optimal parameter values against the 'standard profile'.

Parameter	Average Residual ( $\sigma$ )	Average Residual (%)
Random Walk Step Size Gain	+ 0.86	+ 8.74
Composite Update Interval (s)	+ 0.06	+ 1.16
Composite Correlation ( $\kappa_c$ )	+ 0.25	+ 4.82
Composite Bias ( $\kappa_b$ )	+ 0.32	+ 7.30

Figure 4.13 and Table 4.1 demonstrate the changing optimal parameters, as well as the corresponding difference between the results of the 'standard profile' and swarms using the optimal parameter. Residuals were given by

$$r = \mu_o - \mu_s \quad (4.1)$$

where  $r$  is the residual, and  $\mu_o, \mu_s$  are the mean totals of the optimised and standard results respectively. Where percentage residuals are used, this is given by  $r$  as a percentage of  $\mu_s$ . Where normalised residuals are used, the standard deviation used to normalise was given by

$$\sigma_r = \sqrt{\sigma_o^2 + \sigma_s^2} \quad (4.2)$$

in accordance with the method of propagating uncertainties given by Hughes and Hase [62]. Here,  $\sigma_r$ ,  $\sigma_o$ , and  $\sigma_s$  represent the standard deviations on the residuals, results of optimal parameters, and results of standard parameters respectively.

It should be noted that where an optimal parameter value matches the standard value, the resulting residual can only be attributed to natural variations between tests (such as those seen throughout Section 4.2), as experimental setup was otherwise identical.

The optimal RWG was consistently 1.0 despite tests indicating otherwise for differing swarm sizes and algorithms, this also exhibited by far the largest normalised residual difference from the standard profile, giving normalised residuals of over 2 standard deviations in one case. The percentage residuals are also reasonably high, but not to the same degree as the normalised residuals.

The optimal update rate was reasonably consistent compared to the standard profile and exhibited correspondingly low residuals, only one value gave different results to the standard profile, and resulted in poorer performance.

Both random walk parameters exhibit a considerable range of optimal values which rarely agreed with the standard profile, with no discernible correlation based on the fractal dimension, but, interestingly, a seemingly inverse correlation with each other. The residuals for the  $\kappa_c$  value were generally low (all within 1 standard deviation and less than 10%), while the residuals for  $\kappa_b$  were comparably to  $\kappa_c$  for normalised residuals but very high by percentage. These results indicate that the varying values of  $\kappa_c$  do not impact the performance of the swarm as critically as the variation in  $\kappa_b$ , but also that changes to  $\kappa_b$  exhibits a naturally larger variation in results.

Additionally Table 4.1 demonstrates that there is, on average, an increase in performance for all parameters when values are selected optimally. However, this increase is generally lower than 10% and mostly less than one half of the standard deviation exhibited by the residual, with all being less than one standard deviation. This does not indicate a strong effect of any of the tested values on the swarm's performance, particularly when compared to the effects of changing the swarm size or environment design shown in Section 4.2, where the resulting performance of the swarm can vary by orders of magnitude. This additionally has the effect of vindicating the results of Section 4.2: had the results of Figure 4.13 and Table 4.1 demonstrated that a variation in optimal parameters resulted in a dramatically different outcome for otherwise identical swarms, the validity of the use of the standard profile when testing would have been lessened. The demonstration that the variation of optimally-tuned swarms against a 'standard profile' is less than the natural variation between other variables shows that this method is still valid when testing the

effects of differing behaviours, environments and swarm sizes. This does not mean that attaining optimal values is not worthwhile in practice, as a performance increase is still observed, but that it is not as important as selecting the correct method and swarm size to employ in a specific environment.

# Chapter 5

## Conclusions

### 5.1 Optimal Parameters in Foraging

The findings of Section 3.2.7 demonstrate the difficulty of developing swarm systems that are reliant on pre-tuned parameters. The many different combinations of behaviours, swarm sizes, and environments, result in an overwhelmingly large amount of data required to determine optimal values, even if all of the variables are assumed to be independent, which is unlikely to be accurate.

The variation in optimal parameters shown in Section 4.3 demonstrates that prediction of optimal values is difficult without extensive and exhaustive testing of various combinations and ranges for each parameter; even within a small subset of variables (changing only environment while keeping the behaviour type and swarm size constant) there was a large variation exhibited in the optimal correlation and bias parameters. However, while there can be a large variation in the value of the optimal parameters, the effects on swarm performance are less pronounced. The tests were run only on a small subset of variables, but exhibited a change in performance of generally less than 10%, and were all within a variation of one standard deviation, indicating that much of the change may be attributable to the natural variation. The results did indicate a consistent increase in performance overall when using optimal parameters, and individual tests occasionally exhibited a strong increase, though it must be noted that these values were generally far less than the changes in performance exhibited by variation of environment, swarm size, or foraging algorithm.

Three options are subsequently available:

The first option is the use of a single ‘profile’ which is used for all environments. This will naturally result in a less-than optimal performance for most situations, but simplifies the implementation greatly.

The second option is the development of multiple ‘profiles’, potentially grouped together; this would likely lead to improved performance, but reduces comparability between tests, requires a more in-depth tuning process, and can lead to issues with over-fitting to data, making it unsuitable to applications outside of that which it was trained on. Additionally, the demonstration that optimal parameters have less of an effect on performance than variations in swarm size or behaviour type indicate that many of these profiles are likely to be redundant.

The third option is an on-line optimisation or adaptive algorithm: swarm robotics have potential as a self-optimising system. The use of evolutionary algorithms on swarms is well-recognised [24, 31], and could be considered a natural fit for such optimisation. The main downside to this method is that such on-line methods suffer from a performance lag, and would rely on a pre-existing profile to start from. In real-world scenarios where the swarm is used in foreign and unknown environments, such as search-and-rescue, such a performance lag would be undesirable.

Further investigation is needed to develop the best solution to this issue, but it is this author’s recommendation that parameters be tuned optimally only once the most appropriate swarm size and behaviour has been selected for a task, as they are shown to affect performance more than any parameter, and doing so reduces the dimensionality of the problem greatly. Where the environment of the task is completely unknown, the use of a standard profile followed by an adaptive, on-line optimisation would be the strongest course of action.

## 5.2 Algorithm Comparisons

Random walks, implemented as a baseline study, were occasionally superior to more complex algorithms at very low swarm sizes. The Site Fidelity, or Biased Random Walk exhibited performance that was superior to or on parity with a simple Correlated Walk, and so is recommended as a method of baseline behaviour in all foraging algorithms (in lieu of any additional data that can be acted upon).

It is evident that additional work is needed to improve the performance of the Kasprzok method, as it consistently underperforms compared to all other algorithms, including the Random Walks. The exact cause of this is unclear, but it is likely to be a combination of the issues with translating the system from an idealised model to a simulated swarm. However, as the method represents a means of foraging with relatively low information and could be adapted to work on a passive communication system such as visual-only, it should not be entirely disregarded as a concept. The general performance of the implementation



as is, however, is not worthy of further consideration.

Information-mediated algorithms, namely the Ducatelle and Composite methods, were vastly superior to all others except in some cases using very small swarm sizes. The general trend was that the Composite method was superior at smaller swarm sizes (up to 40 robots) with the Ducatelle out-performing it past this. At higher clutter factors, the Composite method was shown to be comparable enough to the Ducatelle while maintaining a low variability between tests that is considered to be the best foraging algorithm for *all* swarm sizes. There may be benefits in investigating the exact nature of this shift in which algorithm is better in order to develop an algorithm which offers reliably superior performance at all swarm sizes. Further investigations at higher swarm sizes may be beneficial, as in many cases the Ducatelle exhibits more rapidly diminishing returns than the Composite method, it may be that the latter returns to being superior at larger swarms.

As expected, all algorithms gave the best performance in an open arena, followed closely by the low-clutter environment of Arena 2. Despite having a higher Fractal Dimension, Arena 3 offered improved performance to the maze-like Arena 4, potentially highlighting the issues with the use of Fractal Dimension explicitly as a measure of clutter. Additionally, these high-clutter environments also caused the information-mediated method to exhibit much larger variances than lower-cluttered arenas, lowering their reliability. The random arenas were the only example where the Composite method exhibited constant diminishing returns, and also exhibited the highest overall variance for all algorithms. It is worth noting that despite the extremely high clutter factor of the random arenas, the performance of many algorithms was superior to Arenas 3 and 4, supporting the assertion that a more complex measure of how cluttered an environment is than Fractal Dimension alone. Otherwise, the ability for most algorithms, particularly the more advanced ones, to achieve performance close to that of a mildly cluttered environment is encouraging for further investigation in this area.

### 5.3 Composite Method

As well as the implementation of four existing algorithms, the novel Composite method was developed. Designed to address the shortcomings of the Ducatelle method, the algorithm made three main adjustments: only using distance to address the preference of robots when selecting information to make use of, an anti-parallel navigation strategy when the observed density of agents is sparse, and the implementation of a range/distance ratio as an estimate of the degree of clutter in an environment, affecting the amount of correlation and bias the random walk exhibits. These three features, combined, were shown to offer vastly superior

performance to other algorithms at low to medium swarm sizes, and continued to be a strong performer at higher levels, with performance exhibiting both less variance within tests and comparable or less in the way of diminishing returns as a result of increasing swarm size than the Ducatelle method. It is also considered that the use of the anti-parallel navigation method may also introduce a small amount of randomness into the agents' decision-making capabilities, serving as a means of either negative feedback or randomness essential to effective emergent behaviour [6], in this case, allowing agents to explore nearby paths which may not be so congested.

It is currently unclear as to the relative contributions of each feature to the algorithm's performance, as they were only tested in combination. Immediate areas for improvement include acting on the information of all agents rather than those returning from a searcher's goal, and a more in-depth implementation of a density-dependent phase change in behaviour. Finally the range/distance ratio offers potential for developing further work in navigation in cluttered environments, especially if a more complex behaviour reacting to the changing value of the ratio can be developed.

Overall, the performance of the algorithm was very strong, and this author is willing to propose it as an alternative or even an enhancement of the existing Ducatelle method. Given the Composite method's lower variation both between swarm sizes and within tests on swarms of the same size, the Composite method could be confidently considered the best overall strategy of those considered, and one of the best non-stigmergic methods of foraging in cluttered environments.

## 5.4 Improvements and Future Work

While the analysis of the Kasprzok method in this work has been largely unfavourable, the author acknowledges that the system required more adaptation than most of the other methods listed. While every effort was made to emulate the system successfully, and it is entirely possible the method is simply incapable of being applied in its original state to the issue of cluttered foraging, a deeper investigation into improvements to the algorithm, or especially its most favourable characteristics may be merited.

Almost any control system will require the tuning of several parameters; as discussed, the use of foreign environments and varying swarm types results in strong difficulty in generating optimal parameters without exhaustive testing on specific environments and hoping that application to similar environments will be enough to generate acceptable performance. It is recommended that more investigation be conducted into on-line optimisation methods for swarms, in conjunction with the development of optimal 'profiles' which can

be employed to ensure that the performance lag caused by such self-optimising was not overly detrimental, particularly in scenarios where immediate performance is the primary requirement of the system.

As this work was explicitly comparative, there was less of a focus on investigating individual algorithms and their relative strengths. As such, the development of a new foraging behaviour was not conducted with as much rigour as would have been in a work focusing strictly on the development and improvement of such a behaviour. However, the promising initial implementation indicates that this is an area worthy of further exploration: the Ducatelle method is one of the pinnacles of non-stigmergic foraging and as such, the Composite method's ability to address some of its shortcomings represents a strong candidate for further development. The major developments would be investigation of the interplay of the implemented systems, as well as possible development of the Ducatelle's sequence number to act as an indicator of congestion: if two packets of information have travelled a similar distance but with varying numbers of transmissions, it can be assumed that the lower number of transmissions represents a less congested path. This idea is still in its infancy but may act to enhance the performance of the Ducatelle/Composite methods at high agent densities.

Additionally, more investigations at higher swarm sizes may be merited. There is evidence in several experiments, notably those shown in Figure 4.9 and 4.11 indicate that the Composite method could become superior to the Ducatelle method at higher swarm sizes than was tested, due to a difference in the diminishing returns of each algorithm. Further tests at higher swarm sizes may demonstrate that the change in which behaviour is preferable may not be as clean-cut as the current range of swarm sizes generally indicates.

Finally, though the implementation of a method which was completely independent of direct communication was not implemented, this would likely be a useful concept to explore in the future. In line with this, while it was not conducted here, an investigation of the information-mediated methods' performances with integrated noise and information loss may affect the findings here. The ARGoS simulator allows for simulated noise and packet loss from all sensors and communication actuators; or else implementation on real robots would allow for an unrivalled representation of the capabilities of the behaviours. Regrettably, this was not within the scope for this work, but would be essential for further investigation.

## 5.5 Summary

Though there are considerable difficulties in optimally tuning a multi-agent system designed for use in foreign environments, an effective comparison has been made between five foraging behaviours in a series of environments with varying degrees of clutter. The Kasprzok method exhibited extremely poor performance, possibly due to an inappropriate application, while a Site Fidelity or Biased Random Walk was shown to be superior to a simple Correlated Random Walk and is recommended for use as a baseline foraging behaviour to build on. Information-mediated methods relying heavily on inter-agent communication were vastly superior at all swarm sizes except very small (8 agents), and across all environments. The newly-designed ‘Composite’ method offers comparable performance to the established Ducatelle method at high swarm sizes, and superior performance at smaller ones.

The following contributions have therefore been identified as a result of this study:

The emulation, implementation, and application of multiple foraging algorithms in a variety of cluttered environments beyond those which they were originally designed for.

The direct comparison of these algorithms with each other using both a common robot design based on existing non-idealised agents and common performance metrics. This included subsequent analysis of relative strengths and weaknesses of varying swarm sizes operating in a variety of environments.

The analysis of the effects of differing starting conditions and environments on the optimal values of tuned parameters, confirming that there is a heavy variability between optimal values based on environmental condition, but that the resulting variation in the swarm’s behaviour does not correspond to a significant effect on the algorithm’s performance. Additionally, that this variation is generally of a lower magnitude than the variation exhibited between differing foraging behaviours, swarm sizes, environmental types, or even multiple tests under identical conditions.

The development of a largely novel foraging method, primarily designed to address and overcome the shortcomings of the currently premier Ducatelle method. Further testing, particularly with implementation of realistic noise and physical errors, is required, but though the method’s implementation is still in its infancy, it shows promise as a first-rate method for foraging in cluttered and foreign environments.

# Bibliography

- [1] R. Munroe, “New Robot,” 2019. [Online]. Available: <https://imgs.xkcd.com/comics/new{-}robot.png>
- [2] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, “Swarm robotics: a review from the swarm engineering perspective,” *Swarm Intelligence*, vol. 7, no. 1, pp. 1–41, 2013.
- [3] E. Sahin, “Swarm Robotics: From Sources of Inspiration to Domains of Application,” Tech. Rep., 2005. [Online]. Available: <http://www.kovan.ceng.metu.edu.tr>
- [4] F. Ducatelle, G. A. Di Caro, A. Förster, and L. Gambardella, “Mobile stigmergic markers for navigation in a heterogeneous robotic swarm,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6234, 2010, pp. 456–463.
- [5] J. P. Hecker, K. Letendre, K. Stolleis, D. Washington, and M. E. Moses, “Formica ex machina: Ant swarm foraging from physical to virtual and back again,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7461, 2012, pp. 252–259.
- [6] I. Navarro and F. Matía, “An Introduction to Swarm Robotics,” vol. 2013, 2013.
- [7] L. Iocchi, D. Nardi, and M. Salerno, “Reactivity and Deliberation: A Survey on Multi-Robot Systems,” Tech. Rep.
- [8] S. Nouyan, A. Campo, and M. Dorigo, “Path formation in a robot swarm: Self-organized strategies to find your way home,” *Swarm Intelligence*, vol. 2, no. 1, pp. 1–23, 2008.
- [9] F. Ducatelle, G. Caro, A. Förster, M. Bonani, M. Dorigo, S. Magnenat, F. Mondada, R. O’Grady, C. Pinciroli, P. Régnier, V. Trianni, and L. Gambardella, “Cooperative navigation in robotic swarms,” *Swarm Intelligence*, vol. 8, no. 1, pp. 1–33, 2014.

- 
- [10] G. Pini, M. Gagliolo, A. Brutschy, M. Dorigo, and M. Birattari, “Task partitioning in a robot swarm: a study on the effect of communication,” *Swarm Intelligence*, vol. 7, no. 2, pp. 173–199, 2013.
  - [11] E. Buchanan, A. Pomfret, and J. Timmis, “Dynamic task partitioning for foraging robot swarms,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9882. Springer Verlag, 2016, pp. 113–124.
  - [12] J. Turner, “Termites as models of swarm cognition,” *Swarm Intelligence*, vol. 5, no. 1, pp. 19–43, 2011.
  - [13] N. Franks, J. Hooper, M. Gumn, T. Bridger, J. Marshall, R. Groß, and A. Dornhaus, “Moving targets: collective decisions and flexible choices in house-hunting ants,” *Swarm Intelligence*, vol. 1, no. 2, pp. 81–94, 2007.
  - [14] C. P. Bechlioulis and K. J. Kyriakopoulos, “Collaborative multi-robot transportation in obstacle-cluttered environments via implicit communication,” *Frontiers Robotics AI*, vol. 5, 2018.
  - [15] Z. Husain, D. Ruta, F. Saffre, Y. Al-Hammadi, and A. F. Isakovic, “Search in a maze-like environment with ant algorithms: Complexity, size and energy study,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11172. Springer Verlag, 2018, pp. 150–162.
  - [16] A. Chatty, I. Kallel, P. Gaussier, and A. M. Alimi, “Emergent complex behaviors for swarm robotic systems by local rules,” in *2011 IEEE Workshop on Robotic Intelligence In Informationally Structured Space*. IEEE, 2011, pp. 69–76.
  - [17] J. Abbott and A. P. Engelbrecht, “Nature-inspired swarm robotics algorithms for prioritized foraging,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8667, pp. 246–253, 2014.
  - [18] A. Kasprzok, B. Ayalew, and C. Lau, “An ant-inspired model for multi-agent interaction networks without stigmergy,” *Swarm Intelligence*, vol. 12, no. 1, pp. 53–69, 2018.
  - [19] A. Font Llenas, M. S. Talamali, X. Xu, J. A. R. Marshall, and A. Reina, “Quality-sensitive foraging by a robot swarm through virtual pheromone trails,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11172. Springer Verlag, 2018, pp. 135–149.
-

- 
- [20] L. Pitonakova, R. Crowder, and S. Bullock, “The importance of information flow regulation in preferentially foraging robot swarms,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11172. Springer Verlag, 2018, pp. 277–289.
- [21] J. Abbott and A. P. Engelbrecht, “Performance of bacterial foraging optimization in dynamic environments,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7461, 2012, pp. 284–291.
- [22] R. Mayet, J. Roberz, T. Schmickl, and K. Crailsheim, “Antbots: A feasible visual emulation of pheromone trails for swarm robots,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6234, 2010, pp. 84–94.
- [23] L. Pitonakova, R. Crowder, and S. Bullock, “Information flow principles for plasticity in foraging robot swarms,” *Swarm Intelligence*, vol. 10, no. 1, pp. 33–63, 2016.
- [24] J. Hecker and M. Moses, “Beyond pheromones: evolving error-tolerant, flexible, and scalable ant-inspired robot swarms,” *Swarm Intelligence*, vol. 9, no. 1, pp. 43–70, 2015.
- [25] R. Fujisawa, S. Dobata, K. Sugawara, and F. Matsuno, “Designing pheromone communication in swarm robotics: Group foraging behavior mediated by chemical substance,” *Swarm Intelligence*, vol. 8, no. 3, pp. 227–246, 2014.
- [26] H. Sasaki and H.-f. Leung, “Trail traffic flow prediction by contact frequency among individual ants,” *Swarm Intelligence*, vol. 7, no. 4, pp. 1–20, 2013.
- [27] S. Shah, R. Kothari, S. Jayadeva, and S. Chandra, “Trail formation in ants. A generalized Polya urn process,” *Swarm Intelligence*, vol. 4, no. 2, pp. 145–171, 2010.
- [28] S. Garnier, J. Gautrais, and G. Theraulaz, “The biological principles of swarm intelligence,” *Swarm Intelligence*, vol. 1, no. 1, pp. 3–31, 2007.
- [29] A. Khaliq, M. Di Rocco, and A. Saffiotti, “Stigmergic algorithms for multiple minimalistic robots on an RFID floor,” *Swarm Intelligence*, vol. 8, no. 3, pp. 199–225, 2014.
- [30] P. D. Bailis, R. Nagpal, and J. K. Werfel, “Positional Communication and Private Information in Honeybee Foraging Models,” vol. 6234, 2010.
-

- 
- [31] V. Sperati, V. Trianni, and S. Nolfi, "Self-organised path formation in a swarm of robots," *Swarm Intelligence*, vol. 5, no. 2, pp. 97–119, 2011.
- [32] C. W. Reynolds, "FLOCKS, HERDS, AND SCHOOLS: A DISTRIBUTED BEHAVIORAL MODEL." *Computer Graphics (ACM)*, vol. 21, no. 4, pp. 25–34, 1987.
- [33] J. Svennebring and S. Koenig, "Building Terrain-Covering Ant Robots: A Feasibility Study," *Autonomous Robots*, vol. 16, no. 3, pp. 313–332, 2004.
- [34] A. Purnamadjaja and R. Russell, "Guiding robots' behaviors using pheromone communication," *Autonomous Robots*, vol. 23, no. 2, pp. 113–130, 2007.
- [35] D. J. T. Sumpter, *Collective animal behavior*. Princeton, N.J. ; Oxford: Princeton University Press, 2010.
- [36] D. D. M. Gordon, *Ant encounters : interaction networks and colony behavior /Deborah M. Gordon.*, ser. Primers in complex systems. Princeton, N.J. ; Oxford: Princeton University Press, 2010.
- [37] L. Marsh and C. Onof, "Stigmergic epistemology, stigmergic cognition," *Cognitive Systems Research*, vol. 9, no. 1, pp. 136–149, 2008.
- [38] R. Beckers, J. L. Deneubourg, and S. Goss, "Trails and U-turns in the selection of a path by the ant *Lasius niger*," *Journal of Theoretical Biology*, vol. 159, no. 4, pp. 397–415, 1992.
- [39] J. B. Free, *The social organisation of honeybees /John B. Free*, ser. Institute of Biology's studies in biology ; no. 81. London: Edward Arnold, 1977.
- [40] M. Lindauer, *Communication among social bees /Martin Lindauer*, ser. Harvard books in biology ; no. 2. Cambridge: Harvard University Press, 1971.
- [41] *Production ecology of ants and termites / edited by M.V. Brian.*, ser. International Biological Programme (Series) ; 13. Cambridge: Cambridge University Press, 2009.
- [42] B. Webb, "Reactive and Emergent Behaviour IAR Lecture 4," Tech. Rep. [Online]. Available: <http://www.inf.ed.ac.uk/teaching/courses/iar/handouts12/IAR{-}4{-}Reactive.pdf>
- [43] D. Sumpter and J. T., "The principles of collective animal behaviour," *Philosophical Transactions Of The Royal Society B-Biological Sciences*, vol. 361, no. 1465, pp. 5–22, 2006.
-



- 
- [44] Aristotle, *Metaphysics* 8.6.
- [45] V. Sperati, V. Trianni, and S. Nolfi, “Evolution of self-organised path formation in a swarm of robots,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6234, 2010, pp. 155–166.
- [46] D. J. T. Sumpter, R. P. Mann, and A. Perna, “The modelling cycle for collective animal behaviour,” *Journal of the Royal Society Interface Focus*, vol. 2, no. 6, pp. 764–773, 2012.
- [47] A. Winfield, W. Liu, J. Nembrini, and A. Martinoli, “Modelling a wireless connected swarm of mobile robots,” *Swarm Intelligence*, vol. 2, no. 2, pp. 241–266, 2008.
- [48] Y. Zhao, L. Jiao, R. Zhou, and J. Zhang, “UAV formation control with obstacle avoidance using improved artificial potential fields,” in *2017 36th Chinese Control Conference (CCC)*. Technical Committee on Control Theory, CAA, 2017, pp. 6219–6224.
- [49] I. Jang, H.-S. Shin, and A. Tsourdos, “Local information-based control for probabilistic swarm distribution guidance,” *Swarm Intelligence*, vol. 12, no. 4, pp. 327–359, 2018.
- [50] S. Bennati, “On the role of collective sensing and evolution in group formation,” *Swarm Intelligence*, vol. 12, no. 4, pp. 267–282, 2018.
- [51] M. Coppola, J. Guo, E. Gill, and G. C. H. E. D. Croon, “Provable self-organizing pattern formation by a swarm of robots with limited knowledge,” *Swarm Intelligence*, vol. 13, no. 1, pp. 59–94, 2019. [Online]. Available: <https://doi.org/10.1007/s11721-019-00163-0>
- [52] T. Stirling and D. Floreano, “Energy efficient swarm deployment for search in unknown environments,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6234, 2010, pp. 562–563.
- [53] R. Manor and A. M. Bruckstein, “Guidance of swarms with agents having bearing only and limited visibility sensors,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11172. Springer Verlag, 2018, pp. 44–56.
-

- 
- [54] M. Haque, C. McGowan, Y. Guo, D. Kirkpatrick, and J. A. Adams, "Communication model-task pairing in artificial swarm design," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3073–3080, 2018.
- [55] S. E, S. G, P. K. G, and T. D. K. R, "Foraging behavior analysis of swarm robotics system," *MATEC Web of Conferences*, vol. 144, 2018. [Online]. Available: <https://doaj.org/article/6465e3cd28cb44f4be545d26fc0a76e5>
- [56] P. Saeedi, S.-A. Sorensen, and S. Hailes, "Performance-aware exploration algorithm for search and rescue robots," in *2009 IEEE International Workshop on Safety, Security & Rescue Robotics (SSRR 2009)*. IEEE, 2009, pp. 1–6.
- [57] P. Rybski, A. Larson, M. Lindahl, and M. Gini, "Performance Evaluation of Multiple Robots in a Search and Retrieval Task," Tech. Rep., 1998. [Online]. Available: [www.aaai.org](http://www.aaai.org)
- [58] E. O. Wilson, *The insect societies /Edward O. Wilson*. Cambridge [Mass.] : London: Belknap Press of Harvard University Press ; Distributed by Oxford University Press, 1971.
- [59] C. Dimidov, G. Oriolo, and V. Trianni, "Random walks in swarm robotics: An experiment with Kilobots," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9882. Springer Verlag, 2016, pp. 185–196.
- [60] C. Pinciroli, V. Trianni, R. O'Grady, G. Pini, A. Brutschy, M. Brambilla, N. Mathews, E. Ferrante, G. Di Caro, F. Ducatelle, M. Birattari, L. M. Gambardella, and M. Dorigo, "{ARGoS}: a Modular, Parallel, Multi-Engine Simulator for Multi-Robot Systems," *Swarm Intelligence*, vol. 6, no. 4, pp. 271–295, 2012.
- [61] A. Karperien, "FracLac for ImageJ." [Online]. Available: <http://rsb.info.nih.gov/ij/plugins/fraclac/FLHelp/Introduction.htm>.
- [62] I. Hughes, *Measurements and their uncertainties a practical guide to modern error analysis / Ifan G. Hughes, Thomas P.A. Hase*. Oxford: Oxford University Press, 2010.