Proceedings of the 2022 Winter Simulation Conference
B. Feng, G. Pedrielli, Y. Peng, S. Shashaani, E. Song, C.G. Corlu, L.H. Lee, E.P. Chew, T. Roeder, and P. Lendermann, eds.

EFFECTS OF INFORMATION SHARING ON SWARM BASED COMMUNICATION IN DYNAMIC ENVIRONMENTS

Jenny Tran

The University of Adelaide
North Terrace
Adelaide, SA 5000, AUSTRALIA

Peter Fule

Swordfish
North Terrace
Adelaide, SA 5000, AUSTRALIA

Claudia Szabo

The University of Adelaide
North Terrace
Adelaide, SA 5000, AUSTRALIA

ABSTRACT

The use of swarm intelligence aims to aid information sharing in contested and dynamic environments, where network bandwidth is limited and computational resources are overused. In this paper, we examine the effectiveness of a swarm-inspired data ferrying algorithm in the context of a dynamic environment. We simulate a beehive and observe how bees perform as a swarm through exchanging incomplete data items. Our analysis shows improvements of up to 35% when a stigmergic approach is used, showing the benefit of stigmergic information sharing within various communication protcols.

1 INTRODUCTION

In recent years, the term *swarm* has been applied to various areas of study, ranging from biology to robotics engineering. It is derived from the collective behaviour exhibited by entities that aggregate together. Swarm behaviour is generally defined as emergent behaviour (Szabo and Teo 2015; Szabo et al. 2014; Winfield et al. 2005; Mataric 1993) that occurs when individuals follow simple rules without a centralised control system. This phenomenon is often observed in animal groups, particularly when insects gather to forage or migrate. For example, ants can effectively communicate with each other by leaving behind a chemical trail known as 'pheromones' to signify the presence of a food source nearby (Fraser et al. 2021; Fraser et al. 2020). In doing so, ants have the ability to achieve complex tasks as a colony despite being simple agents equipped with limited communication, computation and sensing abilities. Due to the fascinating success of this self-organised system, ant colony techniques along with other animals such as bees, birds and fish have been studied in computer science and robotics. Hence, swarm intelligence is a biology-inspired computer science field that has been able to solve optimisation problems. Stigmergy is a key concept in Swarm Intelligence (SI) focusing on the mechanism of indirect coordination between simple agents by altering their environment (Fraser and Hunjet 2016).

Many systems use information-sharing approaches that often assume flawless conditions, including the presence of strong connectivity and no node failures. However, several challenges can intervene as data is transferred. For instance, communication in tactical environments may be highly contested or completely denied due to foreign interferences. In this particular context, end-to-end connectivity is impossible for information sharing (Hunjet et al. 2018; Szabo et al. 2020). Instead, a delay-tolerant technique via data

978-1-6654-7661-4/22/$31.00 ©2022 IEEE

ferrying is an appropriate solution for maintaining communication. Data ferrying can be described as the process of transferring electronic data to its receivers by physically moving data from one end-user to another via an intermediate platform (Fraser and Hunjet 2016). This process is comparable to paper postal services where international mail can be transported through multiple facilities before reaching its final destination. Although information sharing can be performed by a single node, a centralised approach is more prone to failure if that node becomes disconnected from the system. Therefore, a stigmergic approach can be applied to data ferrying where the agents may only obtain partial information.

In this paper, we examine the effectiveness of data ferrying algorithms in the context of a dynamic environment, where the information shared may be incomplete. We implement a prototype bee hive model that simulates how bees can collect information from outside of the hive and share it with some of the other bees. Each bee collects only partial information and will only be able to pass it on to some of the other bees. The analysis will look at what algorithm for information sharing get the hive the most collected honey and how their performance changes when different environmental constraints appear.

A centralised system is composed of a single master unit that sends out information to each of the other nodes. It is a common practice for organisations to incorporate systems with a client-server model where clients request and receive information from the server. The central unit structure exudes high dependency on network connectivity, which can also pose a major drawback. In the case where the central node loses connection, there is an abrupt failure of the entire system. This can be highly detrimental within a tactical defence environment as communication is the key to preserving situational awareness and allowing timely decision making (Hunjet et al. 2018). Furthermore, information sharing at large distances or in rural environments through a centralised system may be infeasible due to the heavy reliance on connectivity. As a result, there is no guaranteed way of receiving data if the central node fails.

Swarm intelligence and emergent techniques can be integrated into data ferrying algorithms to combat the issues faced in a denied-communication environment. Data ferrying effectiveness can be increased when many nodes work in a coordinated manner, which can be achieved by using a decentralised system. Unlike in centralised systems, a decentralised system is composed of multiple central units where every node makes its own decision. Therefore, the failure of a central node results in a partial system failure. The final behaviour of the system is the aggregate of the decision of the individual nodes. A swarm-based approach to data ferrying enables a fully autonomous system, which means agents operate independently without the need for a control channel (Fraser and Hunjet 2016). Swarm behaviour is obtainable when individuals interact using simple interactions to produce global behaviour without any complex governance. The main benefit of this specific coordination is the ability for individuals to join or leave the swarm and the system will autonomously adapt because its decentralised nature resolves the single-point-of-failure issue. However, there are also limitations and disadvantages with decentralised systems that must be considered. As the distribution of the data comes from various nodes, there is always a limited understanding of the whole system. The process of exchanging partial information between nodes can become problematic when certain information becomes stale over time. This means data exchanged within the dynamic environment may not be up to date because it has already been changed by the time that it had reached the other nodes. While swarm-based information sharing is deemed more reliable in a contested-communication situation, these obstacles should be taken into consideration when examining the effectiveness of the data ferrying algorithm.

To simplify information sharing challenges whilst retaining stigmergy and other swarming properties, a beehive model is employed to abstract swarm-based information sharing that occurs between honeybees.

2 RELATED WORK

Fraser et al. (2021) examine the usefulness of swarming techniques to address data ferrying in a fragmented environment to provide delay-tolerant type communications. The research gathers results from simulations of various network topologies including a modelled application of an amphibious landing. Two data ferrying approaches are investigated: Newton's Cradle and Stigmergic Round Robin.

The research methodology maintained a consistent experimental procedure, running the multi-agent simulator MASON with 20,00 time steps and repeating this 100 times for each approach. The effectiveness of the Newton's Cradle was determined by the time taken to pass a message from node A to node B. The message transfer times were observed, the simulated values gathered aligned closely with the analytical values retrieved from the expected calculations. Using these metrics is a useful indicator for comparing the swarm intelligent information with the coordinated daisy-chain solution. However, there was a lack of measure for the effectiveness of data ferrying, which is important to finding metrics to measure the effectiveness on the bee model prototype. In contrast, the Stigmergic round robin strategy includes nodes as pheromone markers. A limitation of this method is the decay value has not been fully explored for the optimal value. These experiments did not cover dynamic and contested environments.

In (Hunjet et al. 2018), swarming behaviour is split into four classes. Out of the four, Implicit Communication and No Communication are classified as ready candidates to use in tactical defence networks. These classes are further explored in simulation, emulation and physical experiments. The physical test was swarm robotics platforms located within indoor flights facilities and the results demonstrated a strong alignment between data dissemination capabilities with the simulation.

The advantage of this study was the inclusion of different experiment types and the ability to transfer useful metrics from simulation to physical realisation. The MASON simulator was used to simulate pheromone-based data ferrying where interactions between ferries affected the speed at which they were travelling. The simulations were run for 900 steps with the final distance separating ferries and their final speed(s) recorded. A weakness of the method was the lack of ferries included in the experiments because this does not emulate realistic variables and conditions in the average real-life situation and is thus likely to produce more skewed results; the results gathered involved experiments using either only 1 or 2 ferries. This was also carried on to the physical manifestation where future work is stated to scale up the experiments by using an outdoor setting and increasing the node numbers.

Yuce et al. (2013) describes the basic form of Bee's algorithm (BA) along with recently proposed procedures that improve its speed and accuracy. The aim is to implement the enhanced BA to optimize several different optimization benchmark functions. The enhanced version applied a neighborhood size change and site abandonment (NSSA) strategy to the BA. The performance of the NSSA-BA algorithm was measured through an accuracy analysis, average evaluation and t-test, but the effectiveness of information sharing was not considered for dynamic environments.

3 PROPOSED APPROACH

To examine the effectiveness of a swarm-based approach to information sharing, we focus on the following research questions:

1. What are the metrics for measuring the effectiveness of data ferrying within a beehive model?
2. What improvements to data ferrying approach in the beehive model will achieve maximum effectiveness?

To address these questions, we implement a prototype simulation of a beehive model that mimics the foraging behaviour of honeybees. Our model incorporates basic functions drawn from the Bees Algorithm (BA) (Yuce et al. 2013; Pham and Castellani 2009), which is inspired by the natural foraging behaviour of honeybees. It is an optimisation algorithm that performs both an exploitative neighbourhood search in conjunction with a random explorative search. As described in Figure 1, each point in the search space is considered a food source (flower). A population of n agents (bees) randomly searches the space for desired solutions. When a forager bee visits a flower, its quality is evaluated via a fitness function. The BA finds the most promising solutions and selectively explores their neighbourhoods looking for the global minimum of the objective function. Relevant modifications were applied to the algorithm to satisfy the condition of a dynamic environment where information shared may be incomplete. In particular, the algorithm uses a swarm-based strategy to maximise the collected honey. When executing the beehive model, it is important to consider the process of how a bee forages and the way bees communicate their findings to other bees. We observe these behaviours and measure how well the bees are communicating.

3.1 Bee Model

As seen in Figure 1, the model has five main components, namely, Field, Flower Patch, Bee, Bee hive, and Flower Data.

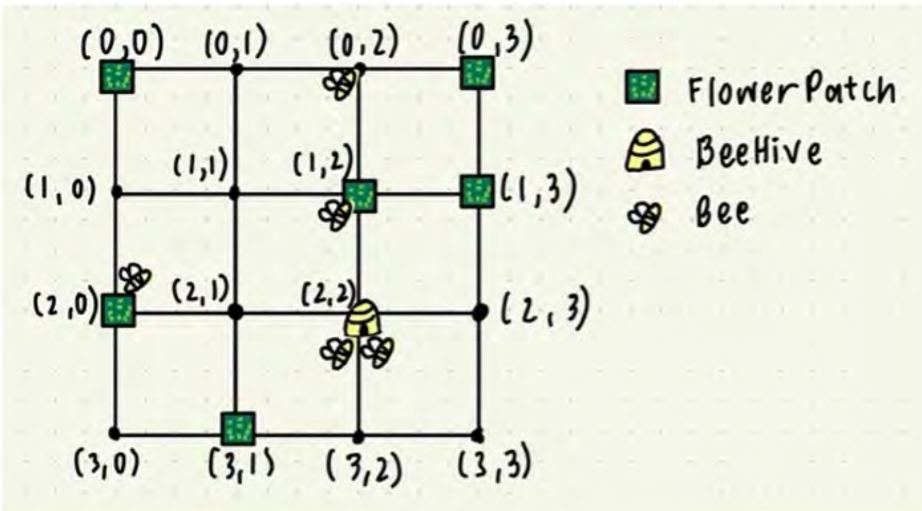

Figure 1: Small-scale diagram of field (4x4) structure,

The behavior of a bee adheres to a state machine that can be seen in Figure 2.

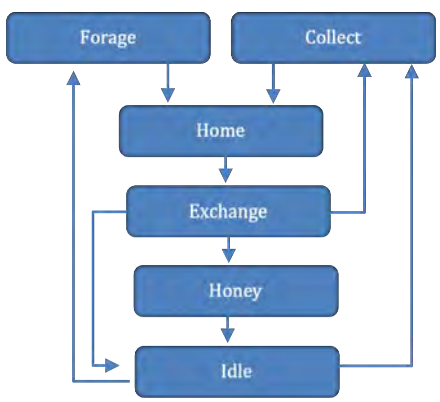

Figure 2: State machine of a bee object.

The states can be described as the following:

Forage: The bee randomly travels to a location in the field. If a flower patch is found, the bee records its data. If not, the bee keeps travelling to a new random location until a patch is found. The bee records the patch's nectar, location and the timestamp of its discovery (tick) in the form of a Flower Data object.

Collect: The bee travels to the best patch according to its fitness function and depletes the patch of its nectar. It continues to deplete from patches until the max nectar capacity is reached.

Home: The bee returns to the Beehive location.

Exchange: The bee exchanges its data collection of patches with other bees that are currently in an idle state. The data is broadcasted to receiving idle bees in the hive. The component of receiving bees denotes a direct form of communication, transferring data to neighbouring nodes. This interaction acts like the bee's waggle dance. The waggle dance is a movement performed by returning forager bees to share information about flower patch sources (Yuce et al. 2013).

Honey: The bee converts x collected nectar into x honey units.

Idle: Bees are on standby at the hive and awaiting their next move (forage or collect). Bees in Idle state receive new information from Exchange bees.

At each time step, all bees undergo the action given by the state machine. The lifespan of each patch is also updated at each tick and removed from the field if it reaches 0.

3.2 Simulation, Data Collection and Analysis

The simulation ran over a number of ticks, storing data of the field at each time step to be used for data analysis. The field was initialised with a grid size of X by Y, which sets the boundary in which the bees and patches reside. A flower patch is initialised with a max lifespan and nectar count. When its lifespan reaches 0, this means that the patch is removed from the field and new patches can also appear. The nectar count of a patch decreases when a bee collects its nectar.

A series of experiments were conducted to observe how altering specific elements in the field would affect information sharing between bees in the hive. In our analysis, we evaluate the total honey production, the total number of completely depleted flower patches, the average bee's data collection accuracy, the average Bee's data collection F1-Score, and the percentage of total bees with at least 5% knowledge of patches in the field. The total honey count is defined as the sum of honey in which all bees produced at the hive. This value is also equivalent to the total nectars the bees collected throughout the simulation. Meanwhile, a completely depleted flower patch is a patch that no longer has any nectar as the bees have fully wiped its sources. Counting the number of depleted flowers is a great indicator of how efficient the data is prioritized among the bees. Both the total number of honey and depleted flower were recorded directly from the data provided by the field list and did not require additional calculations.

Accuracy and F1-score are metrics to test the knowledge of the bee's data collection against all the patches existing in the field. The accuracy measures all the correctly identified cases. Specifically, the sum of true positives (TP) and true negatives (TN) over the total number of cases (including false positives (FP) and false negatives (FN)):

$$accuracy = \frac{TP+TN}{TP+FP+TN+FN}.$$

On the other hand, the F1-score is the harmonic mean of Precision and Recall and gives a better metric for incorrectly classified cases than the Accuracy Metric. It was useful to determine the average bee's accuracy in classifying between a 'Patch' and 'No Patch' in the field:

$$F1\_Score = \frac{TP}{TP + \frac{1}{2*(FP+FN)}}.$$

The average Bee's awareness of the field was obtained by calculating the F1-score and average for each bee's data collection and taking the average from the bee population. Finally, the percentage of total bees with at least 5% knowledge of patches in the field was calculated by tallying the number of bees with knowledge of at least 5% of the field's patches. This percentage of total bees demonstrates the portion of bees that are being exposed to information about patches.

4 EXPERIMENTAL ANALYSIS

In our analysis, we present the average over 10 runs of the dependant variable.

4.1 Experimental Variables

Table 1 displays controlled variables used in these experiments.

Variable	Value
Field Size	35 by 35
Initial Forager Bees	50
Bee Population	500
Total number of Flower Patches	300
Maximum Bee's Capacity	10
Maximum Patch Nectar Count	600
Maximum Patch Lifespan	300
Total Tick	400

Table 1: Controlled variables.

The field size 35x35 means there was a total of 1,225 point locations (0,0,...34,34) in which the bees can travel. The field consists of a beehive, which is initialised with a population of 500 bees where 50 bees commence in *Forage* state while the rest start in an idle state. A bee is only allowed to carry a maximum of 10 nectar units during each Collect state. Furthermore, 300 flower patches are created at random locations in the field with a nectar count restricted to a range of 300-600 and a lifespan of 150-300. The simulation runs over a total of 400 time-steps, updating the patches field as executing bee state function per iteration.

The independent variable is the number of receiving bees during information sharing. A bee will enter the exchange state (Figure 2) after the bee returns to the hive. This bee broadcasts its data collection of patches to some of the other Idle bees that are currently located at the hive. Adjusting the number of receiving bees during the exchange state is hypothesised to impact the production of honey. The independent variable was tested at four different values, namely, 0, 1, 3, 5.

Testing with 0 data receiving bees denotes a No Communication swarm interaction as defined by Hunjet et al. (2018). In this scenario, the swarm is still able to operate individually. As a decentralised system, an individual bee can operate independently on its set of rules and make its own decision. Similarly, adding the component of receiving bees (1, 3, 5) denotes a more direct form of communication, transferring data to neighbouring nodes. This interaction acts as the bee's waggle dance. The waggle dance is a movement performed by returning forager bees to share information about flower patch sources.

4.2 Fitness Function Variations

A variation of the bee's fitness function was adapted to the experiments to improve the effectiveness of information sharing. The fitness function is the process in which the bee determines which patch is the best for sharing information about it.

Basic Fitness Function: Orders bee's data collection by descending nectar count

Improved Fitness Function: Orders bee's data collection by normalised values. The function prioritises additional properties of the patch where the optimal patch consists of the maximum nectar count, the minimum distance from the hive and the most recent timestamp. An offset is added to this value to reduce the effect of bees over flocking patches that would then become depleted too fast[1].

5 RESULTS AND DISCUSSION

In our analysis, we focus on the effectiveness of information sharing strategies at achieving the system goal, in this case, in covering the grid and obtaining useful information from it.

[1]GitHub project repository: https://github.com/jennytrann/swarm-based-information-sharing-simulation

5.1 Effect of Information Sharing on Honey Production

Figure 3 displays the honey production across the simulation run using the basic fitness function. The various levels of information sharing are represented by altering the number of receiving bees during the data exchange. Evidently, the experiments verify that including receiving bees in a stigmergic approach performs better than using a No Communication approach (i.e. 0 number of data receiving bees). However, it is not a constant increase as shown in the higher levels of information. The clustering of these lines suggests that the basic fitness function is not effective in how bees calculate the fittest patch and distributing bees to different locations. Instead, raising the level of information in the swarm leads to similar behaviour across the three levels (1,3,5) where large traffic of the bee population travels to the same exact location only to find there is not enough nectar for each bee. At tick 400, the minimum honey production is 15,451 for zero receiving bees while the maximum is 17,911 for one sharing bee. This means that information sharing using the basic fitness function offered a maximum growth of 15.9% in honey production in comparison to having no communication.

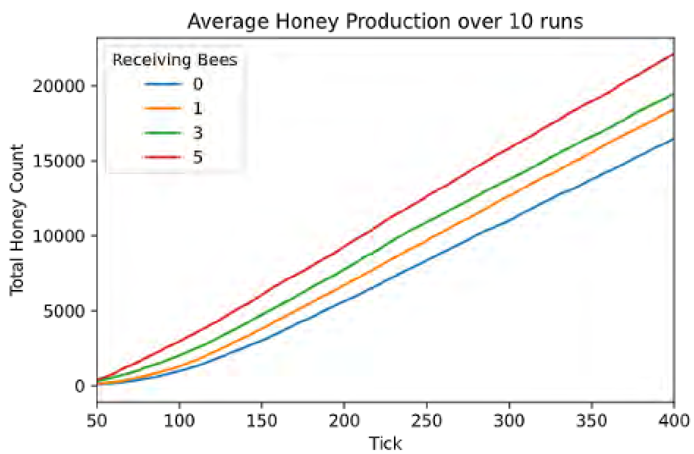

Figure 3: Total honey count vs tick (basic fitness function).

In contrast, the improved fitness function enhanced the performance of the bee's honey production as shown in Figure 4. The emerging trend shows that increasing information sharing also increased honey production. The honey production difference between zero receiving bees and five receiving bees was 5,636, which is a 34% growth from the No Communication approach. By comparing Figures 3 and 4, we can see that the improved fitness function provided a better approach to information sharing in terms of honey production.

Figure 5 shows the number of completely depleted patches across the simulation run using the basic function. A depleted patch is defined as a patch that has been wiped of all its resources and the nectar count reaches zero. This metric represents how the bees' data exchange helps them determine which patch to collect from. As it can be seen, there were no depleted flowers for zero and one receiving bees. Meanwhile, increasing the receiving bees to 3 resulted in an average of 0.7 flowers depleted, and five receiving bees totalled one depleted patch. In a bee simulation with little to no information sharing, there were no patches that are completely exhausted because bees keep data about patches to themselves. Unless another bee finds the same patch in their forage, there is only one bee attending to that patch until it is depleted, and this method is inefficient if collecting from a patch with a large nectar count and a limited lifespan.

The improved fitness function was applied to this experiment and is shown in Figure 6. The results of 0 receiving bees align with the same trend as Figure 5 where there were no patches depleted. However, there was a significant rise in the total number of depleted patches in the higher levels of information sharing. At tick 400, the total rounded number of depleted patches were 3, 5 and 7 for one, three, five

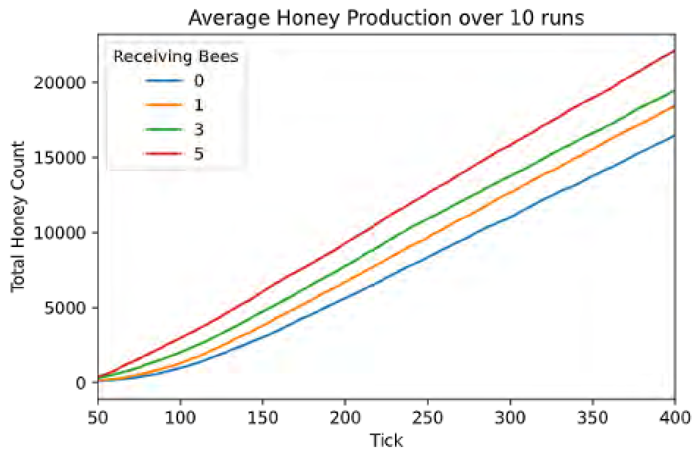

Figure 4: Total honey count vs tick (improved fitness function).

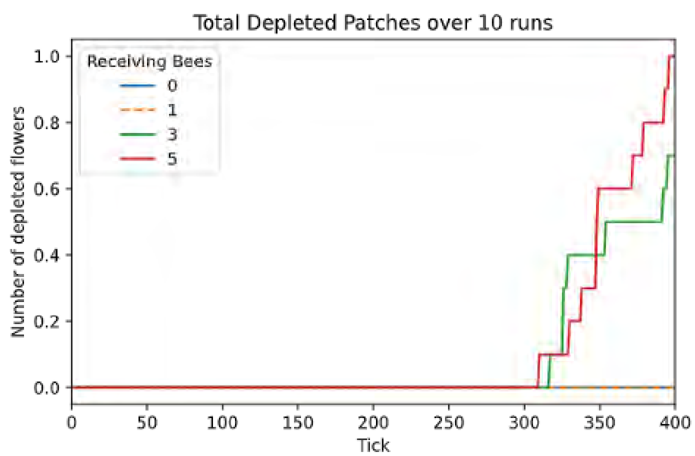

Figure 5: Total number of depleted flower patches vs tick (basic fitness function).

receiving bees respectively. The improved fitness function increased the total depleted flower by a scale of 7 as shown in the data in Figure 5 with a maximum of 1 depleted patch and Figure 6 with a maximum of 7. The increase in depleted flower also reflects the increase in honey production.

5.2 Effect of Information Sharing on Field Awareness

The average bee's awareness of the field was represented by the accuracy and f1-score of the bee's data collection. At the beginning of the simulation, all bees have no information of the field, which can be classified as a prediction of no patches on the field. The bee progressively grows in data while foraging and receiving information from other bees.

Figure 7 shows the accuracy of an average bee's accuracy across the simulation run using the improved fitness function. At the end of the simulation, the accuracy measured 0.75, 0.80, 0.93 and 0.95 for zero, one, three, five receiving bees respectively. We can see for example that an average bee's knowledge of the field increased to 75% accuracy when there are 0 receiving bees.

Figure 8 shows the average's bee data collection F1-score across the simulation run. The F1-score places more emphasis on the correct prediction of recording information about "Patch" rather than "Not patch". The overall trend of the data shows that increasing the number of receiving bees also increases the

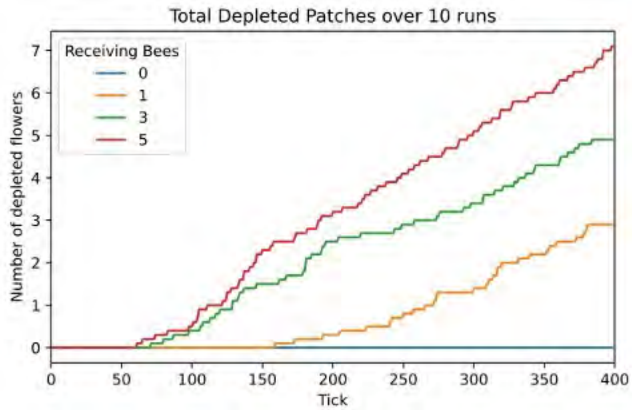

Figure 6: Total number of depleted flower patches vs tick (improved fitness function).

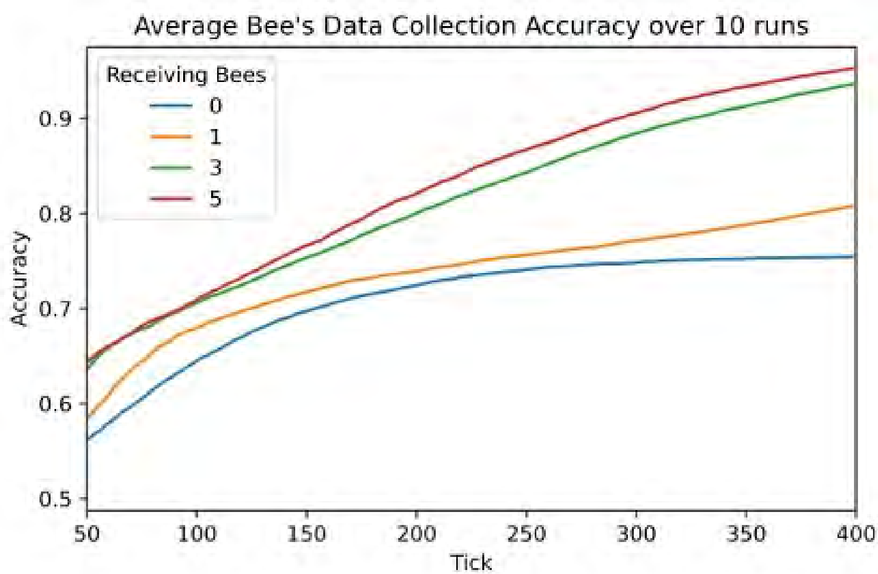

Figure 7: Average bee's data collection accuracy (improved fitness function).

f1-score. At tick 400, the results show an f1-score of 0.01, 0.34, 0.78, 0.86 at information sharing level 0, 1, 3 and five receiving bees respectively. The maximum value of 0.86 signifies that an average bee is 86% accurate in identifying patches in the field. Meanwhile, the minimum value, namely, 0.01, indicates that the average bee has correct data on roughly 1% of the patches. This is equivalent to approximately knowing 3 patches out of 300 patches.

It can thus be deduced that increasing the number of receiving bees during data exchange increases an average bee's knowledge of the field. This is evident in the highest level of information sharing scoring 95% accuracy and 86% f1-score.

5.3 Rate of Information Sharing

Figure 9 displays the percentage of bees with at least 5% knowledge of the patches in the field. This data observes the rate of information sharing during the simulation. Specifically, counting how many bees know at least 15 patches out of the 300 patches in the field. The data states that the population percentages that

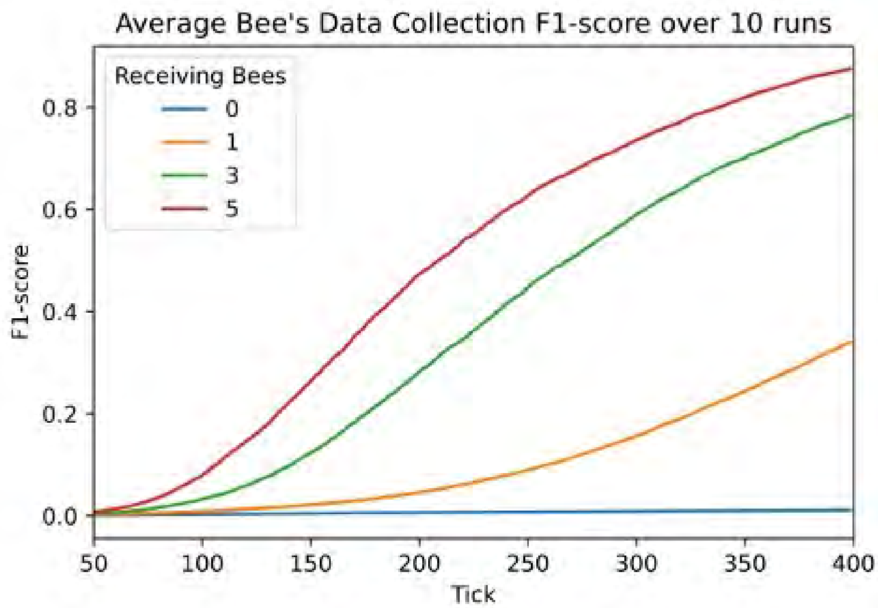

Figure 8: Average bee's data collection F-score (improved fitness function).

satisfy this criterion are 0%, 75%, 94%, 97% at the respective experiment with 0, 1, 3 and five receiving bees. As the number of receiving bees increase, the accuracy becomes closer to approaching 100%. Adding one receiving bee in the simulation spiked a large surge from 0% and 75%. The average bee in a beehive will never reach above 5% when there is no data ferrying that occurs in between the bees and this has a direct impact on nectar collection.

5.4 Discussion

We measured the effect of data ferrying against several relevant metrics. The values include the accumulative honey production and depleted flowers from two versions of the fitness function. Furthermore, the accuracy and f1-score of the average bee's data collection were explored to understand a bee's knowledge of the field.

All experiments tested on the condition with no data ferrying (i.e. zero receiving bees). When testing with this value, it was observed that a bee can operate sufficiently and produce honey without the need for a central control system. A bee makes its own decision based on its set of rules and direction set up by its state machine. Although there is no communication during the exchange state, these bees work independently towards the same goal. The best-case scenario would be if each bee finds a unique patch to collect nectar from for the entire duration of the simulation, so that there is no overlap. However, this is rare and the factor of having a dynamic environment means that the field can change suddenly and information about a patch can become outdated. Our analysis shows how the lack of data ferrying results in a very limited understanding of the field and, consequently, the ability to collect nectar. Specifically, the data shows that no data ferrying produced the lowest honey count, depleted flower count and the lowest average bee's data collection f1-score of 0.01.

In addition to examining an approach with no data ferrying, the same metrics were measured for three different levels of data sharing including one, three and five receiving bees. With five receiving bees, the effect of data ferrying on the beehive simulation increased the average bee's data collection f-score from 0.01 to 0.86 where 97% of the population possessed at least 5% of the field's data. Introducing data ferrying to the experiments made a significant impact on honey production. Foraging and Collecting nectar

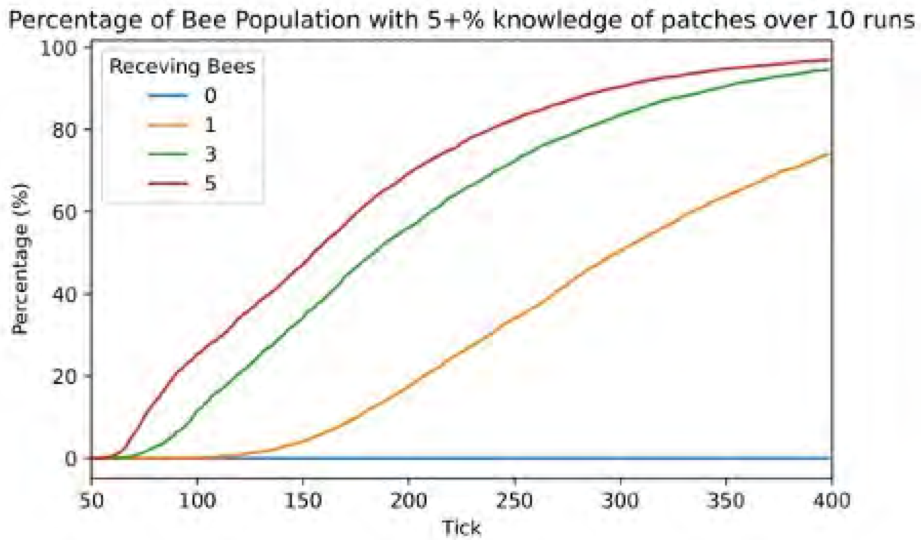

Figure 9: Ratio of bees with at least 5% knowledge of the data.

is computationally expensive as the costs are dependent on the distance travelled by the bee. Hence, the basic fitness was limited in the way it ranked the patches, which resulted in an influx of bees at one patch at a time. The improved fitness function boosted the performance of data ferrying and indirect coordination amongst the beehive, which is evident through the 34% increase in honey production at the highest level of information sharing (five receiving bees) in comparison to no information sharing. Therefore, incorporating a higher number of data receiving bees in combination with the improved fitness function generated the most effective data ferrying approach.

To improve honey production on the beehive model, changes were made to the initial fitness function that sorted a patch's fitness based on its nectar count. The improved fitness function additionally examined the patches by the normalised weighting of nectar count, distance from the hive and its recorded timestamp. Furthermore, a small offset was randomly chosen to reorder the top patches. In doing so, traffic to the fittest patch is reduced and bees are more evenly distributed towards other viable patches. The fitness function can further be refined by adjusting the weighting of the patch properties or adding other factors such as how close the patch is relative to other patches in the field.

Another possible improvement that can be made to the data ferrying approach can be applying a method of choosing which specific bees are involved in the data exchange based on their characteristics. A potential suggestion is to include a selection of bees that have a special characteristic. For example, bees that can travel twice the speed or bees that can carry collect a larger amount of nectar during one trip.

6 CONCLUSION

The adaption of data ferrying in a swarm context allows simple agents to operate individually via a set of basic rules without the need for a central control system. As a collective, the nodes in a decentralised system can accomplish complex tasks through indirect coordination. Swarm-based information sharing was integrated in a prototype bee simulation to observe the way bees interact with the dynamic field of patches and exchange partial information with other bees. The dynamic aspect of the field brought forward multiple challenges to the bees in their interactions with the field. These challenges include acting on stale information and visiting patches that no longer have nectar. Nonetheless, data ferrying offered many positive impacts on the beehive model as seen in the results. Data ferrying provided a significant increase in a bee's awareness of the field and its data accuracy. In return, the bees effectively use a swarm-based data ferrying approach to optimise flower patch depletion and honey production. In expanding the application of this study, swarm-based information sharing is proven to perform effectively in a dynamic environment and its versatility should be examined on other models such as defence communication systems and robotics.

REFERENCES

Fraser, B., and R. Hunjet. 2016. "Data Ferrying in Tactical Networks Using Swarm Intelligence and Stigmergic Coordination". In *2016 26th International Telecommunication Networks and Applications Conference (ITNAC)*, 1–6. Dunedin, New Zealand: Institute of Electrical and Electronics Engineers, Inc.

Fraser, B., R. Hunjet, and A. Coyle. 2021. "A Swarm Intelligent Approach to Data Ferrying in Sparse Disconnected Networks". *International Journal of Parallel, Emergent and Distributed Systems* 36(6):630–651.

Fraser, B., C. Szabo, A. Coyle, and R. Hunjet. 2020. "Leveraging Emergent Specialization in a Heterogeneous Multi-Role Swarm Control Architecture for Positional-based UAS Missions". In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2584–2591. Canberra, ACT, Australia: Institute of Electrical and Electronics Engineers, Inc.

Hunjet, R., B. Fraser, T. Stevens, L. Hodges, K. Mayen, J. C. Barca, M. Cochrane, R. Cannizzaro, and J. L. Palmer. 2018. "Data Ferrying with Swarming UAS in Tactical Defence Networks". In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 6381–6388. Brisbane, Australia: Institute of Electrical and Electronics Engineers, Inc.

Mataric, M. J. 1993. "Designing emergent behaviors: From local interactions to collective intelligence". In *Proceedings of the Second International Conference on Simulation of Adaptive Behavior*, 432–441.

Pham, D. T., and M. Castellani. 2009. "The Bees Algorithm: Modelling Foraging Behaviour to Solve Continuous Optimization Problems". *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* 223(12):2919–2938.

Szabo, C., B. Sims, T. Mcatee, R. Lodge, and R. Hunjet. 2020. "Self-Adaptive Software Systems in Contested and Resource-Constrained Environments: Overview and Challenges". *IEEE Access* 9:10711–10728.

Szabo, C., and Y. M. Teo. 2015. "Formalization of Weak Emergence in Multiagent Systems". *ACM Transactions on Modeling and Computer Simulation (TOMACS)* 26(1):1–25.

Szabo, C., Y. M. Teo, and G. K. Chengleput. 2014. "Understanding Complex Systems: Using Interaction as a Measure of Emergence". In *Proceedings of the Winter Simulation Conference 2014*, 207–218. IEEE.

Winfield, A. F., J. Sa, M.-C. Fernández-Gago, C. Dixon, and M. Fisher. 2005. "On formal specification of emergent behaviours in swarm robotic systems". *International Journal of Advanced Robotic Systems* 2(4):39.

Yuce, B., M. S. Packianather, E. Mastrocinque, D. T. Pham, and A. Lambiase. 2013. "Honey Bees Inspired Optimization Method: the Bees Algorithm". *Insects* 4(4):646–662.

AUTHOR BIOGRAPHIES

JENNY TRAN is an undergraduate researcher in the Complex Systems Research Group at the University of Adelaide. Her email address is jenny.tran03@student.adelaide.edu.au.

Peter Fule is a Principal Software Engineer for Swordfish Computing. His email address is peter@swordfish.com.au.

CLAUDIA SZABO is an Associate Professor in the School of Computer Science and the Lead of the Complex Systems Research Group at the University of Adelaide. Her research interests lie in the area of complex systems and on using simulation to identify and validate their emergent properties. Her email address is claudia.szabo@adelaide.edu.au.