A Robot-Based Simulation of Honeybee Dance Communication

Beibei (Lily) Dai



4th Year Project Report Artificial Intelligence and Computer Science School of Informatics University of Edinburgh

2024

Abstract

This project presents a study on the emulation of the waggle dance of *Apis mellifera* using robotics, aiming to capture the core of this complex bee communication method. This research focuses on a dancer robot that performs path integration during foraging to return to the hive and subsequently executes a simplified dance to communicate the location and distance of food. A follower robot, observing the dance, decodes the signals to derive a vector, enabling it to locate the food source. A practical realisation of the waggle dance was achieved in a robotic context, with the simulation system achieving accuracy within an average deviation of 3.334° and a distance error of 0.146 meters. The work navigated technical challenges, including sensor integration and mechanical replication of biological motion, and assessed their impact on the fidelity of the simulation. These findings offer insights into the potential of robotics to simulate animal behaviour, providing a groundwork for future applications in current robotic systems and contributing to the broader understanding of bio-inspired technology.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(Beibei (Lily) Dai)

Acknowledgements

I would like to express my deepest gratitude to all those who provided their invaluable support and insights during the completion of this project.

Firstly, my sincerest thanks to my supervisor Professor Barbara Webb, for her unwavering guidance, patience, and expertise. Without her knowledge and encouragement, this would not have reached its fruition. Additionally, I am deeply thankful for being given the opportunity to undertake a self-initiated Honours project in my field of interest.

I am also immensely grateful to Anna Hadjitofi, Robert Mitchell and Evripidis Gkanias, who offered help and suggestions in the development of this work. Their perspectives and diligence have greatly enhanced the quality and depth of this project.

I would also like to acknowledge the assistance of Garry Ellard and Katy Lobban, who provided technical assistance and hardware materials. Their support was invaluable and greatly appreciated.

Last but certainly not least, I wish to express my profound appreciation to my family and friends (Amy Yin, Yuhan Cheng, Jiayin Guo, Ruoxi Liu, and Zhuoning Wu) for their understanding, patience, and endless love. Their moral support during this journey was my anchor and provided me with the strength to persevere.

There are still so many people not mentioned here, but they all contributed to completing this project. This will be an unforgettable experience in my undergraduate life.

Table of Contents

1	Intr	oduction	1
2	Lite	arature Review	3
	2.1	Introduction of Dance Communication	3
	2.2	Purpose and Benefit of Dance Communication	5
	2.3	Mechanism of Dance Communication	6
		2.3.1 Prerequisite: Path Integration on Central Complex Model	6
		2.3.2 Encoding Waggle Dance by Dancers	8
		2.3.3 Decoding Waggle Dance by Followers	9
3	Met	hodology	12
	3.1	Hardware Experimental Setup and Scale	12
	3.2	Experimental Procedure and Software Architecture	14
	3.3	Dancer Bee TurtleBot	15
		3.3.1 Foraging Procedure	15
		3.3.2 Dance Circuit	17
	3.4	Follower Bee TurtleBot	18
		3.4.1 Antennae-inspired Algorithm for Orientation of Dancer	18
		3.4.2 Decoding the Resource Information	20
	3.5	Summary	21
4	Res	ults	22
	4.1	Path Integration Robustness	22
	4.2	Dance Communication Robustness	24
	4.3	General System Robustness	26
		4.3.1 Resource Distance	27
		4.3.2 Direction of Resource Angle	28
		4.3.3 Magnitude of Resource Angle	29
	4.4	Overall	31
5	Disc	cussion and Conclusion	32
	5.1	Challenges	32
	5.2	Further Study	34
	5.3	Conclusion	35
Bi	bliog	raphy	37

A N	Matl	nematical Description for CX Model	41
A	A .1	Firing Rate Model	41
A	4.2	Layer 1 - Speed Input and Directional Input	41
A	4.3	Layer 2	42
A	4.4	Layer 3 - Compass	42
A	4.5	Layer 4 - Speed accumulation	43
A	4.6	Layer 5 - Steering Output	43

Chapter 1

Introduction

At the intersection of biology, cognitive neuroscience, and robotics, this project investigates the fascinating waggle dance of honeybees, showcasing the interplay between complex communicative behaviours in insects and the application of biomimetic principles in robotics. Central to the study is the waggle dance, a sophisticated form of communication among honey bees that conveys information about the distance, direction, and quality of food sources through a series of symbolic movements. The foundational discoveries by Karl von Frisch have expanded the understanding of animal communication and opened new avenues for technological innovation, particularly in the design and implementation of autonomous robotic systems that mimic biological intelligence and functionality.

This interdisciplinary research focuses on bridging the gap between the biological processes underlying dance communication and their implementation within the robotics field. A cornerstone of this project is the development and refinement of computational models for path integration and dance communication. These models are inspired by the neurobiological mechanisms of bee cognition, specifically targeting the central complex (CX) of the bee brain, which plays a pivotal role in spatial orientation and navigation. By translating biological insights into algorithmic frameworks capable of execution by robotic platforms, the project aims to replicate the sensory processing, spatial navigation, and communicative capacities of honey bees in a robotic context.

Embarking on this project presented a multitude of technical and conceptual challenges. On the technical front, integrating sensors and actuators to mimic the cues from the dance required a solid understanding of both hardware capabilities and biological functions. Conceptual barriers, involving the abstraction of biological intelligence into computational algorithms and the simulation of highly adaptive and fluid natural behaviours in mechanical systems, were equally challenging.

Despite these challenges, significant steps were made in performing the waggle dance within robotic systems. Through programming and hardware integration, a model for path integration that closely mirrors the bee's ability to navigate and store resource position was successfully implemented. Furthermore, this project can act as an interactive educational platform, merging biology, robotics, and computer science to inspire

Chapter 1. Introduction

cross-disciplinary exploration. By using robots to simulate bee dances, it engages the public in complex biological communication in an accessible way. The development of bee cognition models aims to enhance cognitive science, providing a controlled setting for testing theories on memory, learning, and communication, thus linking theoretical concepts with practical observations.

In conclusion, this project comprises the following subtasks:

- Implemented ROS 2 control on TurtleBots.
- Reimplemented an existing neural network for path integration.
- Created a local coordinate system backup to prevent robots from erroneous movements.
- Programmed hive recognition
- Created a waggle dance behaviour model.
- Designed a multi-sensory system for the follower robot.
- Implemented the extraction of relative angle estimations.
- Applied mathematical model converting relative angles to resource direction.
- Established sound detection.
- Conducted multiple full-cycle tests on the system.

Chapter 2

Literature Review

2.1 Introduction of Dance Communication

The waggle dance of the honey bee (*Apis mellifera*) has been extensively studied as one of the most complicated and fascinating examples of a non-human communication system in nature, captivating scientists from various fields for decades (Frisch, 1993; Dyer, 2002; I'Anson Price and Grüter, 2015; Barron and Plath, 2017; Dong et al., 2023). In the dance, foraging bees share the location and quality of food sources outside the hive with fellow nestmates, as well as potential nesting sites (Frisch, 1993). This remarkable behaviour represents unique evidence of sophistication in animal language and its significance extends beyond its biological novelty, offering insights into the cognitive and communicative capabilities of these social insects.

The waggle dance was first brought to the attention of the scientific community by the pioneering work of the German scientist Karl von Frisch in the mid-20th century, who unveiled various aspects of dance language and was awarded the Nobel Prize in Physiology or Medicine in 1973 (NobelPrize.org, 2023). His groundbreaking research discovered that the waggle dance is a form of symbolic communication within invertebrates, exclusive to the genus *Apis*, and a mechanism through which bees convey detailed quantitative distance and directional information of



Figure 2.1: Schematic diagram of the waggle dance of *Apis mellifera* (Barron and Plath, 2017)

resources relative to the hive using a series of highly stereotyped movements in a figureof-eight pattern (Frisch, 1993). Its distinctive 'waggle run' involves wing vibrations and abdomen waggling from side to side as the bee moves forward in a straight line on the comb inside the dark hive upon successfully finding resources (Dyer, 2002). The duration of the waggle phase indicates the distance of the resource from the hive, while the orientation of the waggle phase relative to gravity or the direction of the sun indicates the direction of the resource (Gardner et al., 2008; Schürch et al., 2013). The mechanism of dance communication will be described in detail in Section 2.3.



Figure 2.2: Inputs to the central complex, which play a role in orientation, are also believed to be involved in the dance (Barron and Plath, 2017)

Beyond performing the dance, understanding how bees obtain information about resource locations before communicating it via the dance is important for appreciating the complexity of this system. The underlying neurobiological processes involve gathering, memorising, and encoding spatial information from the environment. Comparative analyses within the genus *Apis* and among other social insects (El Jundi et al., 2014; Pfeiffer and Homberg, 2014; Heinze and Homberg, 2007) have shed light on the CX in the bee's brain, a key area involved in spatial orientation and navigation. It plays a critical role in cognitive tasks, processing visual cues and optic flow from bees to respectively provide compass information and distance information (Brockmann and Robinson, 2007; Stone et al., 2017). Furthermore, Barron and Plath highlight the adaptive significance of the waggle dance, proposing it as an evolved behaviour that leverages pre-existing neural mechanisms for spatial navigation (Barron and Plath, 2017). In detail, the CX region could help connect the processing of environmental information to dance communication (Figure 2.2). This suggests that the waggle dance is not just a unique form of communication but also a reuse of the bees' inherent navigational abilities.

Early research indicates that recruits or followers, who acquire information about resources from waggle dances, learn from the dancer bees through a sophisticated integration of sensory information, including visual, vibratory, and olfactory cues, to decipher the dance's symbolic message (Rohrseitz and Tautz, 1999; Dyer, 2002; Seeley, 2011). This requires recruits to be in close proximity to the dancer (Gardner et al., 2008), tracking her through numerous waggle phases to decode the resource's location with precision. The dance also communicates the quality of resources through the dancer's pheromonal signals and the specific scents of the resources (Galizia, 2014), enhancing foraging efficiency. Hadjitofi and Webb propose that particular neural circuits in the central complex brain region could be used to decode the dancer's directional cues, with tactile feedback from antennal contact allowing followers to assimilate information

from various positions (Hadjitofi and Webb, 2024). This highlights the complexity of how bees interpret dances to find resources, showcasing a complex relationship between behaviour and brain function in navigating and foraging activities.

2.2 Purpose and Benefit of Dance Communication

The waggle dance of honey bees is not just a mesmerising spectacle but also a highly evolved form of communication, essential for the survival and efficiency of bee colonies (Dornhaus and Chittka, 2004; Dornhaus et al., 2006). Its key advantage is guiding bees to abundant, valuable food sources without the need for individual exploration. This form of efficient recruitment is especially valuable in environments where floral resources are scarce, hidden, or highly variable (Sherman and Visscher, 2002; Seeley, 2012). Through dance communication, a colony can swiftly converge on newly discovered feeding sites, greatly improving foraging efficiency and, consequently, the overall energy intake of the colony.

Research indicates that the effectiveness of the waggle dance is context-dependent, providing the most significant benefits in habitats with particular characteristics (Dornhaus and Chittka, 2004; Toufailia et al., 2013; Nürnberger et al., 2017). In tropical forests, for instance, where floral resources are often clumped but ephemeral, the waggle dance enables bees to quickly exploit these resources before they vanish (I'Anson Price and Grüter, 2015). This ensures efficient dissemination of location information throughout the colony, reducing the energy and time expended on unproductive searches. Moreover, the waggle dance serves as an adaptive response to the diverse ecological challenges bees face. In regions where food sources are widely dispersed and hard to find, the dance is invaluable for sustaining the colony and facilitating communication about resources that might otherwise be overlooked due to individual foraging limitations (Sherman and Visscher, 2002; Seeley, 2011).

Empirical studies also underscore the dance's role in selecting foraging targets (Dornhaus and Chittka, 2004). Bees tend to perform the waggle dance more for higher-quality resources—those with greater energetic value or essential nutrients (Seeley and Towne, 1992). This selective communication focuses the colony's efforts on the most advantageous foraging opportunities, optimising the balance between collective energy expenditure and intake. Additionally, the dance promotes a dynamic and responsive foraging strategy, enabling a bee colony to adjust to changing resource landscapes (Dornhaus and Chittka, 2004). This adaptability is particularly essential in temperate regions, where seasonal changes significantly impact the availability and distribution of floral resources.

The evolution of the waggle dance is believed to have been driven by the need to efficiently locate and exploit floral resources. Originating from an ancestor likely nesting in open environments, honey bees developed this unique communication form (I'Anson Price and Grüter, 2015). The precise timeline of its evolution is uncertain, with some estimates dating it between 40 million and 20 million years ago, around the time when extant honey bee species diverged during the early Miocene (Engel and Schultz, 1997; Nieh, 2004; Engel et al., 2009). Fossil records and phylogenetic analyses suggest

a potential origin in Europe or Asia, in climates ranging from sub-tropical to temperate, with patchy vegetation (Kotthoff et al., 2013). Initially, the dance may have evolved to aid foraging in these settings, where finding and utilising clustered, high-quality food sources was essential for colony survival.

2.3 Mechanism of Dance Communication

2.3.1 Prerequisite: Path Integration on Central Complex Model

From Section 2.1, the foraging of honeybees starts with a common strategy known as path integration to calculate their homeward route, which utilises neural circuits in the CX area (Srinivasan, 2015). This is achieved by continuously integrating the distances covered in each direction during their outbound journey, forming a home vector containing the direction facing the hive. Unlike some insects that rely on a mix of self-generated motion cues (Seelig and Jayaraman, 2015), bees primarily use visual information for path integration. They utilise celestial skylight cues, such as polarised light, for directional guidance and optic flow for distance estimation (Stone et al., 2017).

In detail, honeybees possess specialised regions in their eyes that detect polarised light, which relates to the sun's azimuth. They have the ability to encode heading directions within a global reference frame for navigation purposes during foraging (Heinze and Homberg, 2007). Research on the tropical nocturnal bee, *Megalopta genalis* (Warrant et al., 2004), shows that neurons in the CX region exhibit a significant response to polarised light. Stone and colleagues identified these neurons as TB1 neurons, located within the protocerebral bridge, and as CPU1 neurons, in the upper division of the central body (Stone et al., 2017), based on the earlier discovery of locusts' compass neurons (Vitzthum et al., 2002; Warrant et al., 2004). Their activities support the conclusion that the CX operates as an internal compass for bees, facilitating their navigation.

Besides compass information, acquiring speed information is also essential for path integration. This is achieved through the interpretation of translational optic flow (Srinivasan, 2014), a form of visual streaming across bees' retinas when they move continuously in one direction. Then the neurons in the CX encode this visual movement information. Studies involving *Drosophila* and cockroaches (Weir et al., 2014; Kathman et al., 2014) indicate the presence of CX neurons sensitive to large-field motion cues. Specifically, in the honeybees' brain, noduli tangential neurons (TN neurons) show strong responses to translational optic flow (Stone et al., 2017). These neurons, with subtypes TN1 and TN2, exhibit differential responses: TN2 neurons show increased firing rates proportional to the velocity of optic flow in a forward direction, whereas TN1 neurons form the groundwork for representing holonomic movements, in which the bee's body axis is not aligned with its direction of movement, along its flight trajectory (Stone et al., 2017). Such responses allow TN neurons to act as a neural odometer in complex environments.

The path integration circuit utilised in the current project was first introduced by Stone



Figure 2.3: Principal connections of all cell types from the proposed path integration circuit in the CX region, from Figure 5F Stone et al. (2017)

et al. (2017), featuring five layers of neurons specifically designed for processing visual information (Figure 2.3). In 2019 (Le Moël et al., 2019), the model underwent further refinements with the additional layer for Vector-Memory and its recalibration. This update introduces a hypothetical memory neuron that uses synaptic weights for vector storage, effectively inhibiting CPU4 integrator cell outputs. Each CPU4 output fibre is linked to sixteen inhibitory synapses, with their weights calibrated based on the activity of the CPU4 fibre at learning instances, potentially indicated by a reinforcer neuron. ICPU4 values are processed through. a sigmoid function that excludes instantaneous noise, aiming for an accurate reflection of CPU4 activity. Learning is initiated by specific events, such as discovering food and facilitating navigation back to the locations where vectors were stored. With all necessary inputs provided and a new feature incorporated, the subsequent paragraphs will sequentially describe the CX model, layer by layer.

The first layer of the CX model receives compass inputs from TL neurons and speed inputs from TN neurons, as outlined in previous sections. Layer 2 inversely processes outputs from TL neurons to generate inputs for CL1 neurons, which are the columnar cells of the lower division of the central body (ellipsoid body), thereby linking compass inputs to the protocerebral bridge (PB). These inputs are thought to directly influence the TB1 neurons in Layer 3, which are tuned to specific azimuth directions, effectively segmenting the azimuthal space around the honeybee. The interaction among TB1 neurons suggests a ring attractor model, which establishes a stable and distributed sinusoidal pattern. Here, the activity of TB1 neurons represents the bee's current heading direction.

Layer 4 contains a collection of integrator cells named CPU4 neurons that process information regarding the bees' speed from TN neurons. Their input is modulated in

conjunction with TB1 neuron activity, resulting in the accumulation of distance data in the direction opposite to the current heading. This process constructs a representation of the home vector, delineating the path home. In the final stages of navigation, CPU1 neurons, serving as steering cells in Layer 5, are engaged. In the updated version of the model, the inputs of CPU1 depend on the state of the bee: only as the honeybee is homing, they evaluate the activation of CPU4 (target direction) against that of the current direction cells to ascertain the correct turning angle. Otherwise, the steering cells update themselves. This sophisticated system supports complex navigation and homing mechanisms and even provides the basis for dance communication.

2.3.2 Encoding Waggle Dance by Dancers

Following successful foraging, this section explores the role of biological mechanisms of dance communication. Most honeybees, such as Apis mellifera, primarily perform their waggle dances on vertical combs within dark nest cavities, using the orientation of the waggle phase relative to vertical gravity to signal the direction of resources relative to the solar azimuth upon departure from the hive as Figure 2.4 B and C (Frisch, 1993). In these instances, the CX model must not only process celestial, landmark-based, or proprioceptive information but also incorporate information on gravity, as gravity is used as an orientational reference frame (Barron and Plath, 2017). To deal with gravitational information, geosensing in honeybees is potentially located in the neck, where the inclination of the thorax alters the pressure exerted on it by the head, influencing geotactic behaviour and dance orientation (Lindauer and Nedel, 1959; Frisch, 1993). While other potential geosensing sites exist, such as the thorax-abdomen joint and leg joints (Srinivasan, 2011), their roles in dance behaviour and their integration into the



Figure 2.4: Diversity of dance and nest forms, Figure 1 from l'Anson Price and Grüter (2015)

CX network require further exploration by the community.

In this project, we focus on dances performed on the horizontal surfaces on the top of the comb at the hive entrance, characteristic of dwarf honeybees such as Apis and reniformis and Apis florea in Figure 2.4 A (Barron and Plath, 2017). These species construct their nests as a single sheet of comb suspended from a tree limb. For their horizontal waggle dances, the waggle run points directly towards the resource, utilising celestial cues (the sky) and landmarks processed by the CX circuit to guide the dance (Dyer, 2002).

The duration of the waggle phase (t in seconds) is a critical component correlating with the distance from the resource to the hive (d in kilometres), fundamentally linked to the concept of optic flow—the visual perception of landscape movement experienced during flight. Recent studies (Gardner et al., 2008; Schürch et al., 2013) confirm that the longer the waggle phase, the further the resource is from the hive, usually showing a positive linear correlation between waggle duration and distance, despite noticeable variations around the linear fit. (Kohl and Rutschmann, 2021) has proposed that the relationship between distance and duration might be more accurately depicted by a non-linear model when foraging distances surpass a certain threshold, thereby advancing and complicating the original concept by (Frisch and Jander, 1957).

Kohl and Rutschmann (2021) segmented the entire dance circuit of honeybees into two parts: the waggle phase and the return phase. It was found that waggle phase duration exhibits a non-linear relationship:

$$t_w = 0.1993 + \left(\frac{2.0018}{0.6717}\right) \times \left(1 - e^{-0.6717*d}\right)$$

achieving an unbiased fit with r = 0.946, which suggests a flattening slope as distance increases, in contrast to the linear model $t_w = 0.4476 + 1.1152 * d$ which resulted in a slightly biased fit with r = 0.935. Similarly, return phase durations increased with distance, described by a relationship $t_r = 1.3712 + 0.5238 * d$. Across the datasets, circuit durations were better explained by non-linear models:

$$t_c = 1.467 + \left(\frac{2.893}{0.6519}\right) \times \left(1 - e^{-0.6519 * d}\right)$$

with an r-value of 0.904, compared to linear models $t_c = 1.8187 + 1.6390 * d$ with an r-value of 0.894, indicating a complex interaction between distance and dance durations. Nonetheless, for practical implementation purposes and due to the minor difference in model fits, linear models are used in the current computations.

2.3.3 Decoding Waggle Dance by Followers

Upon grasping the process of generating dance movements, the subsequent step focuses on an exploration of how honeybees gather resource information from waggle dances. In earlier research, those bees were referred to as recruits. They can track the dancer from multiple positions through various mechanisms, including physical contact with their antennae on the dancer's body (Frisch, 1993; Rohrseitz and Tautz, 1999), substrateborne vibrations, acoustic signals, and airflows generated by the vibrating wings of the dancer. This complex communication allows recruits to decode the dancer's directional and distance cues from following directly behind (Judd, 1994; Michelsen, 2011). Moreover, recruits often need to observe more than one waggle phase to accurately locate the indicated resources (Couvillon et al., 2012; Schürch et al., 2013). Despite variations in individual waggle runs and ongoing debates about the optimal position for following the dance, it is evident that observing multiple waggle phases enhances recruits' understanding of the resource's location (Tanner and Visscher, 2009).

Building on the previous section, the dance primarily communicates the existence and specifics of a profitable food source, including its direction—oriented with respect to celestial cues or gravity for open and cavity-nesting species, respectively—and the



Figure 2.5: Positioning of follower bees and their antennae when following waggle phases, Figure S2 from Hadjitofi and Webb (2024)

distance, as inferred from the waggle phase's duration. The central complex of the bee brain aids in decoding the received dance movements into navigable flight vectors, incorporating direction and distance information and adapting to different spatial reference frames (Barron and Plath, 2017). Additionally, odour cues, fundamental to the dance, furnish recruits with details about the resource's specific floral source, an understanding further enhanced by nectar donation through trophallaxis (Farina et al., 2005).

A recent study by Hadjitofi and Webb (2024) has shed light on the intricacies of honeybee communication, particularly highlighting the role of the follower bees' antennae. It was discovered that honeybees can interpret the waggle dance from multiple positions around the dancer bee (Figure 2.5) using the relative positions of their antennae to disambiguate their angle to the dancer (and thus, food). This finding challenges the earlier belief that bees must be parallel to the dancer to understand the message (Judd, 1994; Michelsen, 2011). The study uses the term 'followers' instead of 'recruits' to describe bees observing the dance, a terminology that will be consistently used throughout the project.

Hadjitofi and Webb (2024) demonstrate that followers can recover the dance vector from any angle to the dancer, thanks to the mechanosensory capabilities of their antennae. This negates the necessity for direct visual observation or precise alignment with the dancer. Video analyses reveal that bees approaching a dancer exhibit a consistent posture, with antennae symmetrically extended and heads stabilised, suggesting a complex interaction beyond mere visual cues. The positioning and angle of the antennae relative to the bee's body provide insights into how bees perceive their spatial relationship to the dancer, enabling them to decode the dance's message from various angles. When positioned to the left of the dancer, the left antenna angles away from the midline as shown in Figure 2.5, while the right antenna remains closer, and vice versa for positions on the right side of the dancer, with a smooth transition observed between these positions.

Furthermore, the study proposes that a circuit in the CX, shown in Figure 2.6, could be adapted to convert spatial information from an egocentric to an allocentric perspective in the context of the dance. In simpler terms, it could convert the dance's orientation from the local frame of reference of the follower to the global frame of reference. Furthermore, similar behaviours were also observed in other insects (Lu et al., 2022; Lyu et al., 2022). This hypothesis builds on the premise that bees are capable of tracking



Figure 2.6: Proposed mechanism to assimilate the home vector using antennae, Figure 2 (Hadjitofi and Webb, 2024)

head direction relative to gravity, and the position of the antennae influences directional processing in the CX (Homberg, 1985). The model then suggests that adjustments of the antennae during the dance modulate neural patterns in the bee brain, enabling the accurate determination of flight vectors toward food, regardless of the observer's position around the dancer. Incorporating a 360° mapping of antennae angles and their modulation of neural activity, the model highlights the system's adaptability and effectiveness, even in the presence of signal variability. The principles outlined in this study are further explored through mathematical simulations, as detailed in Section 3.4, providing a comprehensive understanding of the mechanisms bees use to navigate and communicate.

Chapter 3

Methodology

3.1 Hardware Experimental Setup and Scale



Figure 3.1: Dancer Bee TurtleBot

Figure 3.2: Follower Bee TurtleBot

Microphone

Time-of-Flight

The entire simulation was constructed using two TurtleBot 3 units, which are among the most popular open-source mobile robot platforms. One served as a 'dancer bee' and the other as a 'follower bee'. The 'Burger' model was selected for its smaller size, enabling it to more accurately mimic the diminutive bees. Both robots were equipped with Raspberry Pi single-board computers as their controllers, running Ubuntu Server 22.04 LTS (a Pi 4 for the dancer and a Pi 3 for the follower). Embedded Inertial Measurement Unit (IMU) sensors featuring a 3-axis gyroscope and a 3-axis accelerometer provided sensory information on the robots' position, orientation, and velocity. For the dancer bee (Figure 3.1), additional enhancements included a Logitech WebCam and a Grove Buzzer for hive detection and waggle dance sound simulation, respectively. To determine the relative angle between the dancer and the follower, two time-of-flight (TOF) sensors were installed on the front side of the follower, along with an extra Microphone to capture the duration of the waggle sound (Figure 3.2). Given the unique capabilities of the TurtleBot3, the Robot Operating System 2 (ROS 2) Humble, coupled with Python 3.10, was chosen as the programming framework for this project. In this context, each Python file constitutes a ROS2 node.

Experimental	Natrual	Experimental
Parameters	Scale	Scale
Foraging Distance	4 km	4 m
Waggle Duration	4.0983 s	40.982 s
Return Duration	3.4664 s	34.662 s
Waggle Speed	0.01504 m/s	0.01504 m/s
Honeybee Size	12 mm	14 cm
Single Hexagon in Comb	5 mm	5.83 cm

Table 3.1: Experimental scale for actual foraging distance in 4 km

As shown in Figure 3.3, the experimental field is a rectangular area measuring 5 meters in length and 3 meters in width. At the bottom of the field, a rectangle measuring 1.20 meters by 0.84 meters was positioned to serve as the hive, which was filled with hexagons to mimic the pattern of a comb. The sun's location throughout the experiment was fixed to the south of the hive at an elevation of 30° above the ground, represented by a red arrow in the figure. To commence each subtask of the experiments, both the dancer and follower robot were required to reset their heading direction towards the sun's location. Additionally, five random yellow circles were distributed within the field, representing the resource locations for the bees to explore.

In order to adapt the honey bee foraging behaviours for indoor experimentation, it is essential to establish an appropriate experimental scale that encom-





passes both scaling up and down. This is because the natural foraging distances for bees are normally measured in kilometres, whereas the experimental field accommodates distances in meters. Consequently, the foraging distance inputs were scaled down by a factor of 1000, keeping the same number while converting units from kilometres to meters. Conversely, parameters such as the size of the bees, the duration of their dances, and the size of the hive were scaled up to enhance their accessibility for experimental analysis and to facilitate human observation, particularly when contrasted with the size of honeybee dances in nature. The specifics of the experimental scaling are detailed in Table 3.1, which includes an example where the foraging distance is 4 kilometres, corresponding to the distance of 4 metres between the hive and resource location A in Figure 3.3.

3.2 Experimental Procedure and Software Architecture

The experimental procedure and software architecture designed to mimic honey bee behaviours is supported by five ROS2 nodes: *path_integration* and *waggle_dance* for the Dancer Bee TurtleBot, along with *sun_direction*, *dance_communication* and *follower_foraging* for the Follower Bee TurtleBot (Figure 3.4).



Figure 3.4: Experimental Procedure and Software Architecture

The Dancer Bee TurtleBot starts the process by travelling to a random resource location, controlled manually via keyboard inputs. A CX model accumulates the displacements in each direction to calculate the home vector through the *path_integration* node. This vector is vital in guiding the dancer back to the hive by the shortest path, using a camera that is activated only during the return journey to detect the hive pattern. The *waggle_dance* node encodes the location of the resource relative to the hive into the orientation and duration of the dancer vector. This information is then utilised to carry out a complete dance circuit, which includes the waggle phase and the return run. The dancer robot executes the dance twice to ensure the follower robot captures the necessary information.

On the other hand, the Follower Bee TurtleBot is initially positioned to face the direction

of the sun and utilises the *sun_direction* node to reset the recorded sun's position in the system, which is an essential reference for subsequent calculations. Through the *dance_communication* node, the Follower observes the waggle dance, using TOF sensors and a microphone to determine the dancer's orientation from the relative angle between the robots, as well as to capture the duration of the waggle phase, respectively. This allows the Follower to decode the direction and distance of the resource. In the concluding phase, with the help of the *follower_foraging* node, the Follower navigates to the resource location as indicated by the dance, completing the cycle of communication and foraging behaviour. **The demonstration of the experiment can be accessed by following the link:** https://www.youtube.com/watch?v=NMiQpYrkb1U.

3.3 Dancer Bee TurtleBot

3.3.1 Foraging Procedure

The initial action for the dancer robot is to exhibit foraging behaviour, which is the focus of the *path_integration* node. The central challenge for this node is to operationalise the CX model. This implemented model, extracted from InvertPy, is a Python library devoted to the compilation and execution of computational models for invertebrate neural processing (Gkanias, 2021). In this project, codes from the IvertPy have been specifically modified to align with TurtleBot's hardware for this initiative. It remains operational throughout the entire foraging process. Although its core functionalities are consistent with the principles outlined in Section 2.3.1, the model is distinctively characterised by its real-time application. In contrast to executing a pre-programmed route, the modified CX model processes sensory input and performs calculations simultaneously, thereby equipping the robot with the agility to handle dynamic movements.

Based on the biological mechanisms observed in honeybees, the CX region of their brains only processes visual information. In creating a digital simulator, it is typical to construct a virtual environment that contains environmental cues invisible to humans but detectable by bees, such as polarised light. Replicating polarised light within a laboratory setting presents challenges; therefore, to streamline the visual inputs in such simulations, the optic flow is estimated on the IMU sensor of TurtleBot. This sensor can deliver linear and angular velocities across three dimensions. By subscribing to the "cmd_vel" topic, the live linear velocity along the x and y axes for the robot is captured with ease. Additionally, the quaternion components from the "odometry" topic are converted into yaw rotation in Euler angles. This provides the current heading direction for the robot, analogous to compass orientation. With the velocities and heading direction established, it is feasible to approximate the optic flow as perceived by each of the bee's 'eyes', assuming a standard divergence angle of 45° to either side. This calculated optic flow then informs the speed inputs for subsequent processing within the CX model.

After capturing the speed and compass inputs needed for Layer 1, the following processes adhere to the principles laid out in Section 2.3.1, continuing until Layer 5, which generates the steering output. This introduces a second significant deviation from the original InvertPy code. In the original model (Gkanias, 2021), because the model operated on a pre-loaded route, there was no process for the outbound journeys and CPU1 neurons were only engaged during homing. The enhanced system's ability to process information in real-time requires a responsive approach where the state of the robot dictates the type of CPU1 neurons' inputs. If the dancer robot remains stationary for more than 80 timestamps, it switches from the "foraging" to the "homing" state. Consequently, CPU4 outputs are then incorporated into the computation of CPU1 neuronal outputs. Furthermore, given the experiment's spatial limitations, the robot's movement yields only minor neuronal accumulation, insufficient to produce a viable steering angle for an actual robot. To address this, the memory gain of CPU4 neurons is amplified from 0.0025 to 0.25, and the steering output is scaled up by a factor of 100.

When the dancer robot's status switches to "homing", the system marks its current location as a resource point. At this condition, a local coordinate system is established with the starting point as the origin, aligning the positive y-axis with the direction of the sun. Within this local framework, the angle between the direction of the resource relative to the solar azimuth on departure from the hive is computed geometrically. Additionally, the distance from the resource to the hive is determined using the standard Euclidean distance formula. As a result, these two parameters—angle and distance—are then encoded as the home vector in a text file, which will subsequently be converted into the dance vector in the following phase. This algorithmic approach differs from the method described in Section 2.3.1, where the home vector is recorded using the Vector-Memory Layer of the CX model. The application of this layer within the robot will be explored in Section 5.2.

The subsequent step involves rotating the robot according to the steering angle determined by the CX model. However, trials with various outbound routes revealed a systemic error in the model's performance, which will be elaborated on in Section 4.1. To ensure the dancer can successfully return to the hive for dance communication, a fallback mechanism was implemented. This mechanism computes the robot's target heading towards the hive, factoring in the sun's direction and the relative angle ascertained through geometric calculations. The target heading is the sum of the sun's direction, the relative angle, and π radians (180°). The robot's orientation is then adjusted by comparing the CX model's heading and the geometrically derived heading. If the difference between the two is less than or equal to 0.2 radians (about 11°), the CX model's heading is used. If the deviation is greater, the geometrically calculated heading guides the robot to rotate to the right orientation.

At this stage, the dancer robot's sole requirement is to return to and halt within the hive. Concurrently, the camera thread is activated to identify the hive pattern, operating alongside the CX model. The Logitech WebCam was selected for its autofocus capability, light correction feature, and support for HD resolution, delivering high-quality footage at 720p and 30fps. This webcam is mounted on a 3D-printed stand, angled downwards towards the ground, with the lens surface tilted approximately 5° from the horizon. Image processing is performed by the OpenCV library, which processes the camera's feed to detect hexagons in the hive. This involves converting the image to grayscale, applying a Gaussian blur, and then thresholding to distinguish hexagonal contours based on their geometric shape and size. The detection proves reliable because the threshold adheres to the predefined hive conditions under consistent lighting environments. As a result, the success rate is nearly 99%, except in cases where the dancer fails to enter the hive area entirely. However, variations in lighting conditions can lead to different outcomes, as the camera's performance varies under different environmental conditions. Upon detecting the appropriate hexagonal pattern, a flag is set to indicate the hive has been reached. When this "reached hive" flag is activated, the robot decelerates and continues to move for an additional 5 seconds to ensure entry into the hive, marking the completion of the dancer robot's foraging process.

3.3.2 Dance Circuit

Upon successful foraging, the dancer robot's subsequent task is to perform the figure-ofeight dance pattern with precise orientation and duration. As outlined in the preceding section, the relative angle and distance between the resource location and the hive are recorded within a text file. Consequently, the *waggle_dance* node initiates by retrieving this information from the file.

In the simulation, the dancer robot aims to mimic honeybee dance behaviours that occur on the horizontal plane of the comb inside the hive, due to limitations in robot mobility on vertical surfaces. As previously mentioned in Section 2.3.2, these honeybees orient their dance directly towards the resource, simplifying the robot's task to merely aligning itself with the resource direction, without considering any gravity reference. The robot is initially positioned to face the sun. Subsequently, it adjusts its orientation to match the angle of the resource relative to the hive, as indicated by the text file, ensuring the correct orientation for waggle dance.

The program employs linear and non-linear methods to transform distance data into the waggle phase duration, with the linear model being the default but adjustable to non-linear if needed. To simplify, the waggle phase is depicted as a straight path rather than oscillations towards the resource, improving the follower robot's accuracy in assessing the relative angle. Further details on waggle dance enhancements are deferred to Section 5.2. Additionally, a buzzer mimics the sounds of wing vibrations and abdomen movements during the waggle phase, with its sound duration matching that of the waggle phase. This feature allows the follower robot to determine the waggle phase duration by measuring the sound's length.

In addition to the waggle phase, the complete dance sequence involves the return run—a smooth arc linking the start and end of the waggle phase, allowing the dancer to maintain consistent orientation throughout each waggle. While navigating curves is relatively straightforward for humans, it poses a significant challenge for robots. Typically, executing a curve involves plotting several points along the trajectory, refining the path with an algorithm like Bezier curves, and then employing a suitable PID controller for the robot to follow the path, a process that can be quite complex. Since the return run does not convey information and isn't the primary focus of the study, a simpler half-circle route suffices for the intended purpose. By designating the midpoint of the waggle path as the circle's centre and using the waggle's length as the diameter, calculating the half-circle's length is straightforward. With the return duration derived from the equation in Section 2.3.2, the dancer robot's linear and angular velocities can be determined. These velocities, once published to the "cmd_vel" topic, will guide the

robot along the arc automatically.

The final step involves integrating all components. In addition to loading resource information and adjusting the heading, the action loop encompasses repeating the waggle phase and return run. A complete circuit consists of an initial waggle phase, a return run in one direction, a second waggle phase, and a return run on the opposite side. The starting direction for the return run (either right or left) is determined by the relative angle between the resource and the hive; for instance, if the dancer turns left during the adjustment phase, the first return run will be on the left, and the pattern reverses for the next time. The repetition pattern can be modified in the code, with the default setting to repeat twice.

3.4 Follower Bee TurtleBot

Switching to the perspective of the follower robot, the main challenge lies in determining the orientation of the dancer's waggle phase within its local frame without relying on communication technologies such as Bluetooth or WiFi. To synchronise both robots within the same coordinate system, the follower robot must initially align itself with the sun. This alignment resets the sun's position in its system and matches the follower's local frame with that of the dancer. This functionality is achieved through the *sun_direction* node, which subscribes to the "odometry" topic and performs the same heading direction calculations as shown in the *path_integration* node.

3.4.1 Antennae-inspired Algorithm for Orientation of Dancer

Drawing on insights from Section 2.3.3, the follower bee is capable of discerning its orientation relative to the dancer by the positioning of its antennae and the angle to the dancer. To simulate this mechanism, it's necessary to consider the integration of small, precisely controlled motors that can mimic bee-like movements. Additionally, determining the type of sensory data to collect from the dancer presents a significant challenge. However, from the standpoint of achieving the primary goal—to ascertain the relative angle between the dancer and follower robots—there are various existing solutions. While these solutions employ different sensors, their underlying principles are similar.

The fundamental mechanism involves positioning two sensors along a baseline, maintaining a fixed

gap between them and ensuring they remain stationary throughout the observation process. This approach diverges significantly from the behaviour





of honeybees but is adopted to enhance subsequent calculations. The sensors emit

signals perpendicular to the baseline, and the observed object must present a flat surface perpendicular to the ground. To accomplish this, the dancer robot was enclosed in a cubic shell (Figure 3.6). When the dancer enters the sensors' detection zone, the readings dramatically shift from several meters to a smaller range, specifically within 1.2 meters. Given the hive's dimensions—1.2 meters in width and 0.84 meters in length—the maximum size of the dance pattern is confined within this space. Taking into account that both robots reside within the hive, the threshold for data readings has been established at 0.6 meters, equivalent to half the hive's width. When readings from both sensors fall within this threshold, the system initiates the calculation of the relative angle between the robots based on the variance in the signals' reflection.



Referring to Figure 3.5, by utilising the gap between sensors (70 mm) and the measurement difference from readings, one can employ basic trigonometric functions to calculate the relative angle (θ) between the two robots. The relative angle (θ) is calculated as:

relative angle
$$\theta = \arctan\left(\frac{\text{gap between sensors}}{\text{measurement difference}}\right)$$

A special case occurs when the dancer moves along the direction perpendicular to the follower's heading direction; both sensors will record the same reading, making the measurement difference

.....

Figure 3.6: Dancer Bee TurtleBot same reading, ma with cubic case 0. To prevent inv. the program directly assigns the relative angle as 90°.

ot same reading, making the measurement difference 0. To prevent invalid calculations in this scenario,

Choosing the appropriate sensor is a crucial aspect of this algorithm. The current range of solutions includes various proximity sensors and cameras available on the market. Considering the natural conditions inside a hive, which is completely dark, and the fact that honeybees do not rely on visual cues during dance communication, cameras are eliminated from consideration. Among the different proximity sensors evaluated, the ultrasonic sensor was initially chosen. However, subsequent testing revealed that the ultrasonic sensor's range (15° horizontally) is relatively broad for the experimental setup, with a gap of 70 mm between the two sensors, leading to signal interference and significant errors in the readings. Consequently, the





Time-of-Flight (ToF) sensor was selected for its high accuracy, offering a resolution of 1 mm and an infrared emitter wavelength of 940 nm, making it well-suited for the study. Nevertheless, the default TOF sensors share the same address when connected to only the I²C port in Raspberry Pi, preventing simultaneous readings. To address this issue, an external Python file, *change_tof_address.py*, was created to assign separate addresses to the left and right sensors.

In biological systems, the CX processes the relative angle between the dancer and follower bees, extracting the dancer's orientation (Section 2.3.3). In simulations, this angle is determined through vector calculations. By considering the movement vector of the dancer, the heading direction vector of the follower, and the sun's direction vector, these three vectors form a triangle shown in Figure 3.7. In this specific scenario, the angular difference between the follower robot and the direction of the sun (β) is 90°, which serves as the case study in Section 4.2. From the diagram, the relative angle between the dancer's direction and the sun, α . The equations are as follows:

$$\beta = \alpha + \theta$$
$$\alpha = \beta - \theta$$

Therefore, the orientation of the waggle phase could be computed based on the relative angle from the sensors. However, the relationship between the three angles depends on the quadrant in which β is located, with details illustrated in Figure 3.8 below. Via this, if the dancer's heading aligns to the left of the sun's direction, the resulting figure is positive; if to the right, it is negative. This mirrors anticlockwise and clockwise rotations at the beginning of the dance.



Figure 3.8: Calculations based on different quadrants of the follower's direction

3.4.2 Decoding the Resource Information

The algorithm enables the generation of a singular orientation for the dancer robot. To enhance the accuracy of this orientation, the program calculates a relative angle between the robots at each timestamp, stores them as an array and averages these to refine the computation of the dancer's heading direction. This heading direction precisely indicates the direction towards the resource location. After observing, the follower robot is positioned at the same departure point as the dancer, with its heading aligned with the sun's direction. Similar to the preparation undertaken by the dancer for the waggle dance, the follower will also rotate towards the desired direction before proceeding forward.

The remaining element to determine is the distance between the resource and the hive. This is accomplished with the addition of a microphone, which detects a specific frequency emitted by the dancer robot during the waggle phase and measures its duration. However, the biological method used to obtain the duration will be discussed in Section 5.2. Additionally, the microphone serves as a flow control mechanism within the program, triggering the TOF sensors only when the detected frequency reaches a certain threshold. This aims to prevent the follower robot from making detection during the return phase. Given that the default repetition for the dance circuit is set to twice, the system averages the duration from both instances. This mean is then used to calculate the distance to the resource, utilising the inverse of the equation presented in Section 2.3.2. It is important to ensure that the transformation method chosen for the decoding phase corresponds to the one used during encoding, whether it be linear or non-linear.

Based on the distance from the resource location to the hive, the follower robot is assigned a constant linear velocity along the x-axis. Consequently, the duration of this outbound journey can be calculated. Comparing the total time taken proves more efficient than computing the Euclidean distance at each timestamp. Upon reaching the vicinity of the food resource, defined by a circle with a 0.22 m radius with the resource as its centre, the follower robot will halt, signifying the successful completion of the entire simulation experiment.

3.5 Summary

In this chapter, the fusion of biology, programming, and robotics is extensively explored through the simulation of honey bee foraging and communication behaviours using TurtleBot 3 robots. Comprehensive hardware and software, including Raspberry Pi computers, various sensors, and the experimental environment, were integrated to create a dynamic platform for the robots to mimic the dance communications of bees. Importantly, the methodology proves the biological navigation strategy, facilitates the accurate encoding and decoding of waggle dances, and enables the follower robot to identify and locate resources based on the dancer's movements. This approach supports the potential of applying robotics to biological studies, bridging the gap between two distinct fields.

Transitioning to the next chapters 4 and 5, , the outcomes of the experimental simulations are discussed. This includes an analysis of the methodology's robustness in accurately simulating bee behaviour, discussions on the implications of the findings, and potential areas for future research improvements. The insights gained from the current methodology lay a solid foundation for further exploration, promising exciting advancements in the interdisciplinary field of robotics and biology, offering a new lens through which to understand and replicate the wonders of nature.

Chapter 4

Results

To assess the overall robustness of the system, the experimental setup was designed to evaluate the precision of path integration using the CX model and dance communication, along with the overarching accuracy of the entire process.

4.1 Path Integration Robustness

The series of experiments focused on evaluating the CX model's path integration efficacy across varied complexity levels, utilising Resource A, aligned with the sun and 4 meters from the hive, as a consistent benchmark. This investigation spanned five distinct tasks to explore how path complexity impacts the model's accuracy, measured by discrepancies in target heading directions between the CX model's outputs and geometric calculations. Path complexity was quantified using the Straightness Index (SI), defined as follows

Straightness Index = $\frac{\text{Straight-line distance between two points}}{\text{Actual path length between those points}}$

Given that the direct distance remains a constant 4 meters, an increase in the actual path length results in a lower SI value, indicating a greater complexity of the task.

The assessment of path integration showed that the CX model's highest accuracy was in Experiment 1, with an SI of 0.98, achieving a minimal heading deviation of -2.06° .

Straightness Index of Path	Target Heading from CX Model (°)	Target Heading from Calculation (°)	CX Model/ Calculation	Resource Angle (°)	Resource Distance (m)
1: 0.98	84.72	85.48	CX Model	-2.06	4.01
2: 0.75	77.34	81.67	CX Model	-0.90	4.02
3: 0.66	71.39	73.95	CX Model	-1.37	3.98
4: 0.55	63.81	52.20	Calculation	-3.47	3.95
5: 0.33	20.57	36.07	Calculation	-6.24	3.98

LOBIA / 1 LIAIATIANANIA NATUAAN LIATA INTANYATIAN LIANUATAAAA ANA LIATA / 'AMA	ovity (
Table 4.1. Relationship between Pain Integration Robustness and Pain Comp	exiiv



Figure 4.1: Dancer Robot Path Tracking with five Path Complexity

As path complexity increased, indicated by a lower SI, the model's accuracy declined. With an SI of 0.75, Experiment 2 saw a -0.90° deviation, and Experiment 3, at an SI of 0.66, slightly improved to -1.37°, illustrating the model's adaptability. Experiment 4's -3.47° deviation at an SI of 0.55 emphasized reliance on geometric navigation, while Experiment 5, at the lowest SI of 0.33, experienced the greatest deviation of -6.24°, marking a significant accuracy decrease with complex paths. Nevertheless, the CX model maintained commendable performance as all the traces end within the hive area, highlighting its reliability amid complexity and the pivotal role of geometric calculations in navigation accuracy, despite evident systematic errors.

The analysis of columns 2 and 3 in Table 4.1 reveals a significant relationship between path complexity and the accuracy of path integration using the CX model. This comparison uses geometric calculations as a benchmark, assuming they are accurate unless affected by drifts from IMU sensors from TurtleBot. It is observed that as paths become more complex, the difference in target heading directions between the CX model and the geometric calculations increases. This phenomenon highlights a prevalent issue in robotics: the discrepancy between a robot's intended command and its actual execution. The accumulation of steering errors is more pronounced in paths with numerous curves and loops, resulting in a greater deviation from the intended destination upon completion of the path. Additionally, inconsistencies between sensory input and computational outputs in the model further compound this issue. With the model comprising five layers, the necessary time to process data can introduce delays, exacerbating error accumulation and potentially causing missed or delayed steering adjustments. Generally, as indicated by column 4, the current implementation of 6.95°, enabling effective navigation back to the hive except in more challenging scenarios.

Moreover, an interesting finding is the variation in target heading directions, which ranges from 36.07° to 85.48° across different tasks. Despite these variations, the "dancer" robot consistently manages to return to the hive area in every scenario. This success is likely influenced by the specific environmental conditions of the experimental setup. Near Resource Location A, the area is encircled by large desktop computers with powerful GPUs, creating an unstable magnetic field that affects the IMU sensor data and, by extension, the robot's navigation. This environmental factor is significant when contrasted with the vast natural habitats of bees, highlighting the impact of electromagnetic interference on the experiment. Although this interference skews the IMU sensor readings and contributes to discrepancies in navigation, the CX model demonstrates an ability to adjust for a certain degree of sensor error. However, its overall reliability is undermined in environments with electromagnetic disturbances, suggesting a limit to the model's capability to compensate for sensor inaccuracies.

4.2 Dance Communication Robustness

The system's core relies on accurately conveying location information through dance between dancer and follower robots. To evaluate this, experiments were set up where a follower robot deciphers the dance of a dancer robot, which consistently signals the position of a Resource B, located at a 22.47° angle and 3.10 meters away (Figure 3.3). The follower robots assume ten distinct orientations around the dancer, with their heading direction represented by the angle's sign: negative for clockwise and positive for anticlockwise relative to the sun's direction as shown in Figure 4.2. Each trial keeps the dancer's waggle dance orientation and duration constant when indicating resource B, ensuring consistency.

The absence of data for Positions 1 and 2 in Table 4.2 stems from the design discrepancies between the experiment's robots and real bees. The TurtleBot, designed to emulate bees, has a square shape, in contrast to the elliptical form of bees that communicate observationally using their sides. The algorithm, as detailed in Section 3.4, recognises only signals from the robot's sides as valid for observation. At Position 1 from Figure 4.2 A, the robot's front faces the TOF sensors, leading to inaccurate angle measurements since



Figure 4.2: Experiments for Dance Communication Robustness

this front is not designed for signal detection, complicating the derivation of the correct relative angle due to its perpendicular orientation to the dancer's direction. Similarly, at Position 2, the robot's back, which is also not configured for angle computations, faces the sensors. Thus, with the system's current setup, it's impossible to obtain relative angles for these positions, highlighting a limitation in the robot's design relative to the task requirements.

The remaining eight positions offer valid data for translating the dance pattern into resource location information. Following the methodology outlined in Section 3.4, the first step involves assessing the relative angle between the two robots. Table 4.2 shows that the discrepancy between the TOF sensor readings and the mathematically calculated ground truth varies, from a slight 0.21° in Position 8 to a maximum of 6.3° in Position 4. The mean deviation is 4.01° , which surpasses expectations. This accuracy is partly attributed to the dancer's slow forward motion during the waggle phase, allowing the follower's TOF sensors to maintain stable and consistent readings. Position 8 warrants further discussion due to its mere 0.21° variation from the ground truth. This exceptional accuracy arises when the robots are perpendicular, resulting in a directly assigned relative angle of 90°, as explicated in Section 3.3. Such an orientation ensures a consistent relative angle at 90°. Consequently, after averaging, the error in the relative angle measurement is significantly diminished. Subsequently, the resource angle calculated by the follower is compared to the actual resource angle of 22.47° . Owing to the direct mathematical conversion from the relative angle to the resource angle, this step introduces no additional error. The most accurate results are observed in Position 8, with Position 4 registering the highest discrepancy, following the same outcome.

All positions successfully interpret waggle duration through the measurement of the buzzer sound's length. The most significant discrepancy occurs at Position 7, presenting

Follower	Relative Angle between Robots (°)		Angle between Heading	Resource	Buzzer	Resource
1 USILIUII	Ground	Angle from	and Direction of Sun (°)	Followor (°)	Duration (c)	Followor (m)
	Truth	TOF Sensors	and Direction of Sun ()	Follower ()	Duration (8)	Follower (III)
Position 1	3.97	None	-154.03	None	38.24	3.03
Position 2	12.73	None	35.20	None	38.47	3.04
Position 3	39.38	43.38	-118.15	18.47	37.58	2.97
Position 4	67.78	61.48	-45.33	16.16	37.69	3.06
Position 5	82.53	76.43	-75.0	28.57	39.07	3.10
Position 6	44.05	42.31	-21.58	20.73	40.11	3.19
Position 7	38.24	40.52	60.71	20.19	31.59	2.43
Position 8	89.79	90.0	112.26	22.26	38.96	3.09
Position 9	67.55	61.69	89.98	27.84	38.62	3.06
Position 10	57.05	62.64	145.42	28.06	39.10	3.10

Table 4.2. Dance	Communication	Robustness ha	ased on ten P	Positions of Fr	nllower Ree
TADIE 4.2. Dance	Communication	nuuusiness ua	aseu un len F		JIIOWEI DEE

a variance of 0.67 meters. This deviation arose from an abrupt noise disturbance during experimentation that prematurely halted detection. Consequently, this data point is regarded as an outlier and has been excluded from following processes. Notably, at Position 5 and 10, the follower identifies the resource distance as exactly 3.10 m, mirroring the dancer's signal. The recorded duration of the detected sound span from 37.58 seconds to 40.11 seconds, indicating a spread of 2.53 seconds. Meanwhile, the error in distance measurements for these positions ranges from 0.01 meters to 0.13 meters, a disparity of 0.12 meters. This suggests that the linear correlation between the duration of sound and distance serves to minimise the errors arising from the measurements.

On average, the error in interpreting the resource angle based on the dancer's orientation is about 4.01° , with a mean discrepancy of 0.044 m in determining the resource distance from the buzzer's duration. These figures highlight the system's overall effectiveness and resilience in dance communication. In conclusion, the follower bee is capable of accurately receiving dance vector information from the majority of positions surrounding the dancer.

4.3 General System Robustness

This section describes the evaluation of a comprehensive foraging cycle: A dancer robot discovers a resource and performs a dance at the hive to communicate the location. A follower robot, observing from the same position, then attempts to interpret the dance and return to the precise resource location indicated by the dancer robot. All foraging routes are designed with comparable complexity to prevent any influence on the experimental outcomes.

The experiment designates five resource points across the field (Figure 3.3). Resource A, requiring no angle change, is 4.0 meters away on the sun's direct path. Resources B and C, on opposite sides of the sun's direction and nearly equidistant from the hive—3.1 meters for B with a 22.47° anticlockwise angle and 2.64 meters for C with a -22.35° clockwise angle—are set for testing directional accuracy. Resource D is closer, 1.78 meters away with a slight -9.78° angle, offering a distance variation comparison

with Resource A. The nearest, Resource E, is at 1.03 meters with a sharp 54.47° angle, challenging the system with significant angular differences. Each resource center is marked by a yellow circle of 0.22 meters radius, amounting to 0.15 square meters or 1% of the 15 square meter field. Success is defined by the follower robot stopping within these circles.

Resource	Resource Angle	Resource Distance	Resource Angle	Resource Distance	
Location	from Dancer (°)	from Dancer (m)	from Follower (°)	from Follower (m)	
Resource A	-1.20	4.02	-1.74	3.97	
Resource B	22.47	3.10	26.51	3.09	
Resource C	-22.35	2.64	-25.24	2.63	
Resource D	-7.18	1.78	-9.78	1.77	
Resource E	54.47	1.03	47.87	1.08	

Table 4.3: System Robustness based on five Resource Locations

4.3.1 Resource Distance

This task examines how well the system operates over various distances. Figure 4.3 A illustrates the entire simulation process for Resources A and D. Dancer robots' paths are solid lines, while followers' are dashed, a standard used in later tasks. This figure directly contrasts the pattern sizes for varying foraging distances with overlapping dance pattern inside the hive. Figures 4.3 B and C focus on the details of the dance traces for Resources A and D, underscoring the distinctive figure-eight pattern of the waggle dance. The duration of the waggle phase varies in proportion to the distance of the resource, which in turn affects the size of the dance pattern.

Nevertheless, a noticeable divergence is observed between the two waggle phases for each dance circuit, with the robots exhibiting a drift at the beginning and end points. This could be due to the minimal angles of rotation required for both Resources A and D; identical hardware errors may have a disproportionately larger effect on smaller movements. Additionally, the drift may result from not including the robot's dimensions in the return phase calculation. To enhance accuracy, a live adjustment mechanism could be integrated to monitor the robot's position, ensuring the dancer robot accurately returns to its initial starting location. Moreover, bees in natural settings also display drift in their waggle dances within the hive, suggesting that the robots' performance, to some extent, mirrors the behaviour of actual bees.

Table 4.3 generally indicates minor deviations in angles and distances between dancer and follower robots, falling within the range of acceptable error for effective foraging. Notably, divergent paths are observed for Resource A, potentially due to cumulative minor errors in contrasting directions. Analyzing Figure 4.3 A, the dancer's inbound angle diverges anticlockwise from the target homing direction, while Figure 4.3 B shows the dance orientation diverges clockwise relative to the vertical target dance orientation. When these are combined, the resulting angular difference for Resource A between the dancer and follower is slight, at 0.52°, equating to a minimal 0.14% variation in the table, with the follower halting within the target zone as indicated by



Figure 4.3: A. Path Tracking for Dancer Robot and Follower Robot on Resource Location A and D; B. Waggle Dance Pattern for Resource Location A; C. Waggle Dance Pattern for Resource Location D

the dashed line's end. The follower's travel distance is also marginally shorter by just 0.05 m than the dancer's. For Resource D, the coordination is even closer, with an angular discrepancy of only 2.6° and an insignificant 0.01 m distance difference. These details attest to the system's considerable robustness in managing distance variances, showcasing enhanced accuracy at reduced distances.

4.3.2 Direction of Resource Angle

Exploring the impact of resource angle direction on system precision, robots traverse to Resources B and C, with their paths marked in red and blue respectively in Figure 4.4 A. Figures 4.4 B and C showcase the waggle dances for B and C, highlighting precise execution by the dancer robots. The waggle phase orientations for B are notably precise, demonstrating accurate directionality. The dance for C shows minimal deviation from the optimal figure-eight pattern, as outlined in the Literature Review (Section 2.1). The difference in pattern sizes between B and C stems from a 0.46 m variation in distance to each resource.

Despite the consistency in the dance pattern's direction for Resource B, there is a notice-



Figure 4.4: A. Path Tracking for Dancer Robot and Follower Robot on Resource Location B and C; B. Waggle Dance Pattern for Resource Location B; C. Waggle Dance Pattern for Resource Location C

able discrepancy between the angles when compared to Figure 4.6. This discrepancy results in the paths for Resource B diverging, with the follower robot stopping outside the resource region, which is the only unsuccessful simulation. Table 4.3 indicates an angular difference for Resource B of 4.04° which, although small, becomes more pronounced due to the distance of 3.1 meters; as distance increases, even slight angular errors are magnified for the system. In contrast, the angular variation for Resource C is a lesser 2.89°. Looking back at the Figure 4.4 A, the robot paths for Resource C are more closely aligned, implying more accurate dance communication replication. For both resources, the system decodes distance information with notable precision, showing only a 0.01 m discrepancy. Overall, the system demonstrates the capability to manage resource locations in varying directions with fundamental robustness.

4.3.3 Magnitude of Resource Angle

The study's last part investigates how the magnitude of the relative resource angle impacts the communication system's accuracy. Dancer robots move towards Resources D and E, their paths shown in Figure 4.5 A. Figure 4.5 B and C display the waggle dances for D and E. As the dance for D is as described in Section 4.3.1, this section will focus on analysing the distinct features and challenges of the waggle dance for

Resource E.



Figure 4.5: A. Path Tracking for Dancer Robot and Follower Robot on Resource Location D and E; B. Waggle Dance Pattern for Resource Location D; C. Waggle Dance Pattern for Resource Location E

Figure 4.5 C clearly shows that the waggle dance pattern for Resource E is distinctly different and confined to a smaller area compared to others. This is attributable to the short distance to Resource E, only 1.03 meters, which, when translated into the waggle duration, results in a mere 15-second performance and a forward movement of only 0.2 meters. This movement is relatively small, particularly when considering the robot's size of 0.14 meters. At this level of magnification, the rotations occurring during the return phase are markedly apparent, as all the short arcs are utilised to recalibrate the dancer's heading direction, ensuring it can resume the previous waggle orientation. These arcs are usually inconsequential in other experiments due to their size being vastly overshadowed by the actual dance movements. However, for Resource E, these adjustment arcs significantly affect the visualization of the dance pattern. If these arcs were disregarded, the resultant pattern would bear more resemblance to those observed for other resources.

Beyond the dance pattern, the brief duration also impacts the decoding of resource information, both in terms of the resource angle and distance. The system registers the largest errors observed across all experiments— 6.6° for angle and 0.05 meters for

distance—as indicated in Table 4.3. The short duration limits the follower robot's ability to collect and average more relative angles between the robots. Additionally, it's easier for the follower to inadvertently capture data from the back side of the dancer bee, introducing further inaccuracies into the calculations. For the duration, a shorter duration coupled with the same microphone delay results in a disproportionately larger error, leading to less stable resource distance estimates. Despite these errors, there isn't a significant deviation between the dancer and follower paths, thanks in part to the short distance which mitigates the impact of errors. As seen in Figure 4.5 A, the follower robot ultimately stops within the resource region, demonstrating successful communication. Therefore, evaluating the performance at Resource E reveals that while the system is more accurate with smaller relative angles, such as with Resource D, it still maintains robust performance across a range of relative resource angles.

4.4 Overall

The experimental evaluation of the robotic foraging system revealed significant insights into its capabilities and limitations. Path integration assessed through the CX model demonstrated high accuracy in simple navigation, with precision diminishing as path complexity increased. This highlighted challenges in maintaining accuracy with complex manoeuvres, exacerbated by processing latency and environmental interference. Furthermore, dance communication robustness, examined through the interpretation accuracy between dancer and follower robots, showcased minimal discrepancies in conveying resource locations. Despite design and algorithmic limitations, the system effectively communicated across different scenarios, indicating strong potential in replicating biological communication within robotics.

The system's overall robustness, assessed through comprehensive foraging cycles over various distances and directions, affirmed its capacity to precisely replicate bee foraging behaviour, despite a slight reduction in accuracy with the rise in task complexity. The quantifiable performance of the system, characterised by an average angular error of 3.334° and an average distance discrepancy of 0.146 meters across five specific resource locations, underscores its considerable robustness. These metrics highlight the custombuilt simulation system's effectiveness in emulating complex natural behaviours and maintaining operational integrity even as challenges escalate.

In summary, the study confirms the simulated foraging system's efficacy in acting complex natural behaviours with high navigational and communication accuracy. Future enhancements should aim at refining the system in a more biologically inspired manner and improving performance across complex scenarios. Additionally, conducting repeated experiments could offer deeper insights into the system's performance. This work advances the understanding of robotic mimetics and supports the development of automatic biological systems capable of composite, coordinated behaviours.

Chapter 5

Discussion and Conclusion

To summarise the project, the primary contributions are as follows:

- Implemented ROS 2 control on TurtleBots with both Raspberry Pi 3 and 4 models.
- Integrated an existing CX model for real-time path integration with reliable accuracy within the experimental range.
- Programmed robust hive recognition with high accuracy.
- Created a waggle dance behaviour model with bee-like sounds correlated to the foraging process.
- Designed a multi-sensor operation system to simultaneously capture information for the follower robot.
- Implemented the extraction of relative angle estimations using TOF sensors.
- Applied a mathematical model to convert relative angles into resource direction for different cases.
- Established sound detection using a microphone with considerable accuracy.
- Designed experiments to test the system's robustness.
- Recorded and edited a video demonstration of the simulation.
- Tracked the robots' movements using the Vicon system to map their paths.
- Conducted 3D modelling and printing of sensor supports.
- Performed soldering for sensor assembly.

5.1 Challenges

In general, the self-proposed project presented a significant challenge, requiring a diverse set of skills: a solid understanding of biological neuron models, extensive

familiarity with hardware platforms such as Raspberry Pi and various sensors, comprehensive experience with ROS 2 programming, and the ability to integrate hardware with software seamlessly.

Initially, it demanded extensive time to acquire essential biological knowledge, as exploring insect brain regions diverged greatly from familiar computer science principles. Starting from the dance communication, it is difficult to familiarise with each neuron and its functionality in the neuron system. Moreover, the hardware setup phase was particularly time-consuming. Due to the strict compatibility requirements of the Robot Operating System (ROS), it was necessary to reinstall Ubuntu 22.04 on the Raspberry Pi 3 to match the ROS version on the laptop. However, the Raspberry Pi 3 faced limitations due to its insufficient memory capacity, leading to nearly 87.9% of the storage being used after basic ROS 2 package installations, causing slow system performance. Consequently, a Raspberry Pi 4 replaced the Pi 3 to ensure computational stability on the CX model. This transition, along with mastering the intricacies of path integration and dance communication in bees, took nearly two months.

In the original project proposal, the simulation only considers the dance communication itself. However, upon understanding the mechanism of dance communication, we realised the critical role of the foraging actions, where information shared during the dance is collected. Consequently, we decided to incorporate this aspect into the project to comprehensively simulate the foraging behaviour, leading to a change in the project title. Although this idea was great, transforming the neuron model into an executable programming project presented the greatest challenge. The initial Python package was a complex project that included various neuron models from insects, with the CX model representing only a small fraction of the entire system. This complexity made it unfeasible to implement the whole project on a Raspberry Pi. Identifying the essential functions required a deep understanding of the program's workflow and filtering out irrelevant code, which tested our analytical skills and enhanced our understanding of the model.

The majority of the remaining challenges stemmed from managing various incidents caused by the unpredictable nature of the hardware system. Both the Dancer and Follower TurtleBots experienced system crashes, with the Follower TurtleBot notably ceasing to respond to commands towards the project's conclusion. After unsuccessful troubleshooting attempts, we were compelled to reinstall the operating system and all software to resolve the issue. Fortunately, there were code backups stored on both GitHub and the local laptop, which significantly expedited the recovery process. During this process, the Follower TurtleBot's 16GB SD card was upgraded to a 32GB one to enhance the Raspberry Pi 3's performance.

Additionally, issues related to the connectivity between sensors and the Raspberry Pi were frequently encountered. All sensors discussed in Section 3.3, except the IMU sensors, were selected, compared, installed, adjusted, and tested personally, requiring skills in soldering, 3D modelling, and cabling. For the TOF sensors as mentioned in Section 3.4, since the Raspberry Pi has only one I²C port and the use of a Grove Hat Base, both sensors ended up sharing the same I²C bus address. To read data from both simultaneously, they needed distinct addresses. Owing to time constraints and the inabil-

ity to procure a multiplexer, a simpler approach was adopted: temporarily disconnect the left sensor to reset the right sensor's address to 0x30, then reconnect the left sensor, which used the default address of 0x29. Those issues significantly strengthened our problem-solving skills and steadied our response to unexpected challenges.

5.2 Further Study

The primary objective of this project is to closely replicate the biological behaviours of honeybees. However, due to constraints in the experimental field and time limitations, certain subtasks could be further aligned with biological principles.

The process begins by interpreting sensory information as inputs for the CX model, as outlined in Section 2.3.1. Currently, optic flow for speed inputs is simulated using wheel velocity, but the addition of two cameras, each positioned at a 45° angle on either side of the robot, could capture this phenomenon. Utilising OpenCV functions, such as *calcOpticFlowFarneback()*, would allow for the computation of speed input for TN neurons similar to methods detailed in (Stone et al., 2017). For compass inputs, maintaining the same experimental field, additional ground markers could provide accurate directional data in the world frame. By upgrading the existing camera to recognise these markers, it's possible to refine the heading direction derived from IMU data. Alternatively, integrating more sophisticated magnetic sensors could ensure more reliable readings. Transitioning the experimental field to an open area with sunlight, and employing a polarised camera would greatly enhance the replication of honeybee eye functionality, offering a closer approximation to their natural navigational abilities. These improvements would also increase the accuracy of the CX model, potentially reducing or even eliminating the reliance on geometric calculations.

Further enhancements to the CX model could involve adjusting the gain parameter in the CPU4 layer to elicit a stronger response to movements. Moreover, the approach to storing resource location information could be improved by integrating it into the Vector Memory Layer in the updated CX model, as discussed in Section 2.3.3. As detailed in Section 3.3, using the Python library (Gkanias, 2021), this layer could be implemented following the CPU4 processing and activated when the robot locates the resource. By applying this model, theoretical concepts can be explored within a robotic context, pushing simulations closer to biomimetic practices.

Consequently, this Vector Memory Layer will supplant the current text file as a means to guide dance communication. To mimic honeybee behaviour more closely in the dance circuit, the waggle phase should incorporate constant oscillations along the dance orientation rather than a mere straightforward movement. However, implementing true oscillations poses a challenge for the current TurtleBot platform, given its relatively larger dimensions compared to bees. As a result, it can only execute curved movements instead of precise oscillations. To tackle this challenge, a promising initial approach involves crafting a robot with a more bee-like design, particularly one that can execute a waggle dance by oscillating its main body on a motor while maintaining a stationary base. Given that the final relative angle between robots is derived from an average, the system has the potential to maintain its reliability even after performing the waggle dance.

Angles occurring on either side of the heading direction will average out to a mean relative angle, mirroring straightforward movement. Nevertheless, this modification may still present difficulties for the follower robot in precisely decoding the dancer robot's movements.

Compared to the dancer robot, the follower robot presents opportunities for more comprehensive investigations. This stems from the fact that current follower robot designs are merely inspired by biological mechanisms, rather than directly replicating them as seen with the dancer robot. The TOF sensors, for instance, are influenced by the concept of bees using their antennae to gauge the relative angle between themselves, despite bees not possessing such sensors. These sensors function by emitting and receiving signals at a fixed position on the bee's head, maintaining a constant distance apart, which is not true for honeybees. A more bee-like approach might involve placing the sensor atop a stick, with a small, precise motor at the stick's base (where it attaches to the head) that rotates in response to the sensor's signals, mimicking the antennae's ability to reflect the relative position between two robots. Ideally, there would be two sticks, each positioned at a default angle of 45° from the head's midline, one on each side, to more accurately simulate bee antennae functionality.

Furthermore, the use of a microphone for duration detection is an overly simplified approach. The duration of directional accuracy measurement for assimilated foodward vectors is gauged using the antennae positions of follower bees, beginning when a nestmate qualifies as a 'follower' during the waggle dance and concluding either when these criteria are no longer satisfied or the dance ceases. This method links the length of the integrated vector to the waggle's duration, thus indirectly representing distance (Hadjitofi and Webb, 2024). Consequently, a function should be developed to compute the position of the stick sensor to formulate the vector. Subsequently, the duration and angle information are utilised as inputs for a customised CX model embedded in the follower robot, translating it into actual resource information based on the principles outlined in Section 2.3.3. Although these enhancements for the follower are somewhat speculative, further research and testing are necessary to assess their feasibility and practical application.

Viewing the system as a whole, the integration between each subtask could be improved. At present, both dancer and follower robots are manually positioned at the departure, dance, and observation points. Automating this process could be achieved by placing ground labels on the hive and employing computer vision to identify and track these distinct locations for different states. In conclusion, while the existing simulation system has successfully emulated biomimetic behaviour with notable robustness, there remains ample scope for further exploration and enhancement.

5.3 Conclusion

This project delves into the honeybee waggle dance, integrating concepts from biology, cognitive neuroscience, and robotics to showcase the potential for robots to replicate simple neural behaviours. By creating a simulation system that accurately reflects the dance communication of honeybees, this work lays the groundwork for reproducing

honeybees' foraging behaviour in this robotic system, highlighting the flexibility of such technologies. The primary achievements and future directions include:

- Successful Biomimicry of Waggle Dance Communication: Achieving an emulation of the waggle dance, this project showcases the system's capability to mimic bee path integration and dance communication with minimal deviation in orientation and distance.
- **Innovative Approach to Robotic Design and Programming**: Overcoming various technical challenges, the project introduces novel hardware and programming solutions, depending on recent technology to establish a solid foundation for the study.
- **Pathway for Enhancing Biological Fidelity in Robotics**: It lays out strategies for more biologically accurate simulations, like polarised light navigation and biomimetic antennae, aiming to closely match robotic systems with biological processes and guide future bio-inspired robotics enhancements.

Overall, this dissertation contributes to the biomimetic robotics field, offering new insights into bee communication and aiding in the development of robotics inspired by biological principles. It underscores the importance of cross-disciplinary work in pushing the envelope of what's possible in science and technology, laying a foundation for further exploration of current bio-inspired models and the advancement of autonomous robotics.

Bibliography

- Barron, A. B. and Plath, J. A. (2017). The evolution of honey bee dance communication: a mechanistic perspective. *Journal of Experimental Biology*, 220(23):4339–4346.
- Brockmann, A. and Robinson, G. E. (2007). Central projections of sensory systems involved in honey bee dance language communication. *Brain Behavior and Evolution*, 70(2):125–136.
- Couvillon, M. J., Riddell Pearce, F. C., Harris-Jones, E. L., Kuepfer, A. M., Mackenzie-Smith, S. J., Rozario, L. A., Schürch, R., and Ratnieks, F. L. (2012). Intra-dance variation among waggle runs and the design of efficient protocols for honey bee dance decoding. *Biology open*, 1(5):467–472.
- Dong, S., Lin, T., Nieh, J. C., and Tan, K. (2023). Social signal learning of the waggle dance in honey bees. *Science*, 379(6636):1015–1018.
- Dornhaus, A. and Chittka, L. (2004). Why do honey bees dance? *Behavioral Ecology and Sociobiology*, 55:395–401.
- Dornhaus, A., Klügl, F., Oechslein, C., Puppe, F., and Chittka, L. (2006). Benefits of recruitment in honey bees: effects of ecology and colony size in an individual-based model. *Behavioral Ecology*, 17(3):336–344.
- Dyer, F. C. (2002). The biology of the dance language. *Annual review of entomology*, 47(1):917–949.
- El Jundi, B., Pfeiffer, K., Heinze, S., and Homberg, U. (2014). Integration of polarization and chromatic cues in the insect sky compass. *Journal of Comparative Physiology A*, 200:575–589.
- Engel, M. S., Hinojosa-Díaz, I. A., and Rasnitsyn, A. P. (2009). A honey bee from the Miocene of Nevada and the biogeography of Apis (Hymenoptera: Apidae: Apini). California Academy of Sciences.
- Engel, M. S. and Schultz, T. R. (1997). Phylogeny and behavior in honey bees (hymenoptera: Apidae). *Annals of the Entomological Society of America*, 90(1):43–53.
- Farina, W. M., Grüter, C., and Díaz, P. C. (2005). Social learning of floral odours inside the honeybee hive. *Proceedings of the Royal Society B: Biological Sciences*, 272(1575):1923–1928.

- Frisch, K. v. (1993). *The dance language and orientation of bees*. Harvard University Press.
- Frisch, K. v. and Jander, R. (1957). Über den schwänzeltanz der bienen. Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology, 40(3):239–263.
- Galizia, C. G. (2014). Olfactory coding in the insect brain: data and conjectures. *European Journal of Neuroscience*, 39(11):1784–1795.
- Gardner, K. E., Seeley, T. D., and Calderone, N. W. (2008). Do honeybees have two discrete dances to advertise food sources? *Animal Behaviour*, 75(4):1291–1300.
- Gkanias, E. (2021). Invertpy. https://github.com/InsectRobotics/InvertPy.
- Hadjitofi, A. and Webb, B. (2024). Honeybee antennal positioning data when following dances.
- Heinze, S. and Homberg, U. (2007). Maplike representation of celestial e-vector orientations in the brain of an insect. *Science*, 315(5814):995–997.
- Homberg, U. (1985). Interneurones of the central complex in the bee brain (apis mellifera, l.). *Journal of insect physiology*, 31(3):251–264.
- I'Anson Price, R. and Grüter, C. (2015). Why, when and where did honey bee dance communication evolve? *Frontiers in Ecology and Evolution*, 3:125.
- Judd, T. M. (1994). The waggle dance of the honey bee: which bees following a dancer successfully acquire the information? *Journal of Insect Behavior*, 8:343–354.
- Kathman, N. D., Kesavan, M., and Ritzmann, R. E. (2014). Encoding wide-field motion and direction in the central complex of the cockroach blaberus discoidalis. *Journal* of Experimental Biology, 217(22):4079–4090.
- Kohl, P. L. and Rutschmann, B. (2021). Honey bees communicate distance via nonlinear waggle duration functions. *PeerJ*, 9:e11187.
- Kotthoff, U., Wappler, T., and Engel, M. S. (2013). Greater past disparity and diversity hints at ancient migrations of european honey bee lineages into africa and asia. *Journal of Biogeography*, 40(10):1832–1838.
- Le Moël, F., Stone, T., Lihoreau, M., Wystrach, A., and Webb, B. (2019). The central complex as a potential substrate for vector based navigation. *Frontiers in psychology*, 10:690.
- Lindauer, M. and Nedel, J. O. (1959). Ein schweresinnesorgan der honigbiene. *Zeitschrift für vergleichende Physiologie*, 42:334–364.
- Lu, J., Behbahani, A. H., Hamburg, L., Westeinde, E. A., Dawson, P. M., Lyu, C., Maimon, G., Dickinson, M. H., Druckmann, S., and Wilson, R. I. (2022). Transforming representations of movement from body-to world-centric space. *Nature*, 601(7891):98–104.

- Lyu, C., Abbott, L., and Maimon, G. (2022). Building an allocentric travelling direction signal via vector computation. *Nature*, 601(7891):92–97.
- Michelsen, A. (2011). How do honey bees obtain information about direction by following dances? In *Honeybee Neurobiology and Behavior: A Tribute to Randolf Menzel*, pages 65–76. Springer.
- Nieh, J. C. (2004). Recruitment communication in stingless bees (hymenoptera, apidae, meliponini). *Apidologie*, 35(2):159–182.
- NobelPrize.org (2023). Karl von frisch facts. Accessed 8 Nov 2023.
- Nürnberger, F., Steffan-Dewenter, I., and Härtel, S. (2017). Combined effects of waggle dance communication and landscape heterogeneity on nectar and pollen uptake in honey bee colonies. *PeerJ*, 5.
- Pfeiffer, K. and Homberg, U. (2014). Organization and functional roles of the central complex in the insect brain. *Annual review of entomology*, 59:165–184.
- Rohrseitz, K. and Tautz, J. (1999). Honey bee dance communication: waggle run direction coded in antennal contacts? *Journal of Comparative Physiology A*, 184:463– 470.
- Schürch, R., Couvillon, M. J., Burns, D. D., Tasman, K., Waxman, D., and Ratnieks, F. L. (2013). Incorporating variability in honey bee waggle dance decoding improves the mapping of communicated resource locations. *Journal of Comparative Physiology* A, 199:1143–1152.
- Seeley, T. D. (2011). Honeybee democracy. Princeton University Press.
- Seeley, T. D. (2012). Progress in understanding how the waggle dance improves the foraging efficiency of honey bee colonies.
- Seeley, T. D. and Towne, W. F. (1992). Tactics of dance choice in honey bees: do foragers compare dances? *Behavioral Ecology and Sociobiology*, 30:59–69.
- Seelig, J. D. and Jayaraman, V. (2015). Neural dynamics for landmark orientation and angular path integration. *Nature*, 521(7551):186–191.
- Sherman, G. and Visscher, P. K. (2002). Honeybee colonies achieve fitness through dancing. *Nature*, 419:920–922.
- Srinivasan, M. V. (2011). Honeybees as a model for the study of visually guided flight, navigation, and biologically inspired robotics. *Physiological reviews*, 91(2):413–460.
- Srinivasan, M. V. (2014). Going with the flow: a brief history of the study of the honeybee's navigational 'odometer'. *Journal of Comparative Physiology A*, 200:563– 573.
- Srinivasan, M. V. (2015). Where paths meet and cross: navigation by path integration in the desert ant and the honeybee. *Journal of Comparative Physiology A*, 201:533–546.
- Stone, T., Webb, B., Adden, A., Weddig, N. B., Honkanen, A., Templin, R., Wcislo, W.,

Scimeca, L., Warrant, E., and Heinze, S. (2017). An anatomically constrained model for path integration in the bee brain. *Current Biology*, 27(20):3069–3085.

- Tanner, D. and Visscher, K. (2009). Does the body orientation of waggle dance followers affect the accuracy of recruitment? *Apidologie*, 40(1):55–62.
- Toufailia, H. A., Couvillon, M. J., Ratnieks, F. L. W., and Grüter, C. (2013). Honey bee waggle dance communication: signal meaning and signal noise affect dance follower behaviour. *Behavioral Ecology and Sociobiology*, 67:549 – 556.
- Vitzthum, H., Müller, M., and Homberg, U. (2002). Neurons of the central complex of the locust schistocerca gregaria are sensitive to polarized light. *Journal of Neuroscience*, 22(3):1114–1125.
- Warrant, E. J., Kelber, A., Gislén, A., Greiner, B., Ribi, W., and Wcislo, W. T. (2004). Nocturnal vision and landmark orientation in a tropical halictid bee. *Current Biology*, 14(15):1309–1318.
- Weir, P. T., Schnell, B., and Dickinson, M. H. (2014). Central complex neurons exhibit behaviorally gated responses to visual motion in drosophila. *Journal of Neurophysiology*, 111(1):62–71.

Appendix A

Mathematical Description for CX Model

A.1 Firing Rate Model

Initially, the circuit employed a firing rate model for neurons with the output firing rate r and a sigmoid function of the input I:

$$r = \frac{1}{(1 + e^{-(aI-b))}}$$

where parameters a and b control the slope and offset of the sigmoid. This value is added to a Gaussian noise N(0, σ_r^2), where $\sigma = 0.1$. The resulting firing rate, spanning all layers, is constrained to stay within the range of 0 to 1 through clipping. This constraint ensures that the introduced noise does not exceed the bounds of [0, 1]. The input I equal the weighted sum of the activity of neurons that synapse onto neuron j:

$$I_j = \sum_i W_{ij} r_i$$

A.2 Layer 1 - Speed Input and Directional Input

In Layer 1, two inputs are present: speed and directional inputs. The speed input is attributed to speed-sensing neurons, denoted as TN2 neurons and calculated based on the optic flow sensed from forward to backward motion, while considering the diagonally offset preferred angles of TN cells in the CX noduli within each hemisphere:

$$I_{TN_L} = [\cos(\theta + \phi), \sin(\theta + \phi)] \cdot v$$
$$I_{TN_R} = [\cos(\theta - \phi), \sin(\theta - \phi)] \cdot v$$

where v is the velocity vector of the agent, θ is the current heading direction and ϕ is the preferred angle of a TN-neuron. As TN2 neurons locate their value within the range

0 and 1, they have responses in a positive linearly proportional manner to I_{TN} while having zero reflection to negative flow which is produced by backward motion:

$$r_{TN_2} = \min(1, \max(0, I_{TN}))$$

By assuming the bee is moving in the heading direction, the response of each TN2 neuron is the same with $I_{TN_L} = I_{TN_R} \cos(\phi) v$.

The directional input of the first layer contains 16 input TL neurons, which are sensitive to polarisation, and each of them is located in a preferred direction α in the range of [0, $7\pi/4$]. The inputs follow the equation below:

$$I_{TL} = \cos(\alpha - \theta)$$

A.3 Layer 2

The second layer comprises 16 neurons that receive inhibitory input proportionate to the output of the initial directional input layer. While this inversion of response across the array is not essential, it is included to emulate the characteristics found in CL1 neurons connecting the polarization input to the protocerebral bridge.

$$I_{CL1} = -r_{TL}$$

A.4 Layer 3 - Compass

There are 8 TB1 neurons situated in Layer 3 within the protocerebral bridge of the CX. Each of these neurons receives input from a pair of CL neurons with matching directional preferences. Their role is to establish mutual inhibitory connections with one another, creating a ring attractor circuit. The input to these TB1 neurons is defined as follows:

$$I_{TB_1} = W_{CL_1, TB_1} r_{CL_1} + W_{TB_1, TB_1} r_{TB_1}$$

where W_{CL_1,TB_1} is a [0,1] matrix converting the pairs of CL neurons to single TB1 neurons, and W_{TB_1,TB_1} indicates to the matrix of inhibitory weights between TB1 neurons where:

$$W_{TB_1,TB_1} = \frac{d(\cos(\alpha_i - \alpha_j) - 1)}{1}$$

 α_i and α_j are the preferred directions for their respective TB1 inputs, besides, d = 0.33 is a scaling factor for the inhibition under the comparison of the CL1 excitation.

A.5 Layer 4 - Speed accumulation

The fourth layer comprises 16 neurons associated with the CPU4 cells located in the upper region of the CX body. These neurons receive inputs from the TB1 neurons and TN2 neurons and accumulate the heading direction through the inhibitory compass modulation of the speed-sensitive neurons.

$$I_{CPU_{4_t}} = I_{CPU_{4_t-1}} + acc \times (r_{TN2_t} - r_{TB1_t} - decay)$$

with r_{TN2} as the speed-sensitive response and r_{TN1} as the compass-sensitive response; acc = 0.0025 and decay = 0.1 resulting in the relative rate of memory accumulation and memory loss.

A.6 Layer 5 - Steering Output

Layer 5 consists of 16 neurons that receive inputs from Layer 3 and Layer 4 regarding the information of the compass and home vector. These inputs can be activated or deactivated depending on the state of the bees. Compass input is inhibitory, aligned with Layer 3 to Layer 4 connections. Steering neurons, identified as CPU1, are linked with these connections, providing a steering signal for the bees. In the "exploring" state, the left and right activity is identical, thus no effect on steering. In the "homing" state, a comparison between compass and integrator cell (CPU4) heading occurs with an offset pattern. In the "using vector-memory" state, the vector-memory balances the integrator, causing the agent to navigate toward the feeder location. Upon reaching the food, the search pattern begins. Equations for different states:

$$I_{CPU_1} = \begin{cases} W_{TB1,CPU1}r_{TB1}, \text{ when exploring} \\ W_{TB1,CPU1}r_{TB1} + W_{CPU4,CPU1}r_{CPU4}, \text{ when homing} \end{cases}$$

where $W_{CPU4,CPU1}$ is the connectivity matrix from CPU4 to CPU1 cells.