Natural optimization: An analysis of self-organization principles found in social insects and their application for optimization

Der Fakultät für Mathematik und Informatik
der Universität Leipzig
eingereichte

DISSERTATION

zur Erlangung des akademischen Grades

DOCTOR RERUM NATURALIUM
(Dr. rer. nat.)

im Fachgebiet
Informatik

vorgelegt

von M.Sc. Konrad Diwold
geboren am 7. März 1982 in Wien, Österreich

Die Annahme der Dissertation wurde empfohlen von:

Prof. Dr. Martin Middendorf, Universität Leipzig
Prof. Dr. Bernd Meyer, Monash University, Australien

Die Verleihung des akademischen Grades erfolgt mit Bestehen
der Verteidigung am 9. Februar 2012 mit dem Gesamtprädikat magna cum laude.

Selbständigkeitserklärung

Hiermit erkläre ich, die vorliegende Dissertation selbständig und ohne unzulässige fremde Hilfe angefertigt zu haben. Ich habe keine anderen als die angeführten Quellen und Hilfsmittel benutzt und sämtliche Textstellen, die wörtlich oder sinngemäß aus veröffentlichten oder unveröffentlichten Schriften entnommen wurden, und alle Angaben, die auf mündlichen Auskünften beruhen, als solche kenntlich gemacht. Ebenfalls sind alle von anderen Personen bereitgestellten Materialen oder erbrachten Dienstleistungen als solche gekennzeichnet.

Leipzig, 12.4.2012

Konrad Diwold

Acknowledgements

When starting a PhD one is, at least that's what I hope, genuinely excited about science. Lots of ideas buzz in one's head regarding the topic one will work on in the next years. As these ideas are not always top-notch, it takes good guidance and feedback from supervisors and co-workers to both encourage the pursuit of good ideas as well as point out the flaws of the not-so-good ones. In this respect I would like to thank my supervisor Martin Middendorf for his support and encouragement during my PhD. I would also like to thank the current and previous members of the Complex and Parallel Systems Group in Leipzig, who have been great colleagues. In addition I would like to thank all the members of the HFSP Project "Optimization in natural systems: Ants, bees and slime moulds", especially Madeleine Beekman and David Sumpter for the pleasant and fruitful stays I had in Australia and Sweden, as well as their feedback and support regarding my research. Special thanks to Alexander Scheidler for the bib-style. Last but not least I would like to thank Cliodhna for being wonderful.

This work was funded by HFSP Grant No. RGP0051/2007-C.

Contents

1. Introduction

> It is interesting thus to follow the intellectual truths of analysis in the phenomena of nature. This correspondence, of which the system of the world will offer us numerous examples, makes one of the greatest charms attached to mathematical speculations.
>
> Pierre-Simon Laplace, *Exposition du systéme du monde*, (1799)

Identifying and mimicking concepts underlying natural phenomena and applying them to solve problems in fields such as computer science, material science and engineering, has grown into a research field itself. As Carver Mead once stated "..engineers would be foolish to ignore the lessons of a billion years of evolution." (Wooley and Lin., 2005, p.1).

Computer science and biology share a long history. Alan Turing and John von Neumann, who can be considered among the founding fathers of computer science, both showed an interest in the theoretical aspects of biological phenomena such as pattern formation (Turing, 1952) and self–replication (von Neumann, 1966). With the exponential increase in computational power over the last decades, computer simulations have become an important tool for studying theoretical concepts underlying the behaviour of biological systems and have been proven to be extremely helpful in the understanding and verification of hypotheses based on empirical observations.

Biological concepts, on the other hand, can often be abstracted and fed back into computer science in the form of new computational paradigms. So-called nature inspired-computation has given rise to concepts which are almost ubiquitous in computer science to day such as neural networks (Anderson and Rosenfeld, 1998), evolutionary computation (Eiben and Smith, 2003) and swarm intelligence (Bonabeau et al., 1999).

In the last years animal collectives such as fish shoals, bird flocks and social insect colonies have received increased attention from the computer science community. This is due to the fact that these systems are able to accomplish very complex tasks without any form of central control. Whether observing huddling penguins, swarming locusts or foraging ants, one always wonders which rules underlie this seemingly complex behaviour as such groups lack central control.

Individuals in such animal collectives usually operate on very simple local rules, while the observed complexity in behaviour arises via the individuals' behaviour on a group level. Since group behaviour does not depend on single individuals, such systems are very robust in terms of fault tolerance as well as being adaptive to change in their dynamic environments.

This thesis tries to cover both aspects described above, namely the use of computational models to investigate open questions regarding the organization and behaviour of social insects, as well as the abstraction of concepts found in social insects to generate new methods in the context of optimization. The first part of this thesis studies different aspects of division of labour via response threshold models. The second part of the thesis focuses on honeybees. Using theoretical models, aspects of the honeybee nest-site selection process as well as migration will be investigated. Based on the nest-site selection process observed in honeybees, an optimization scheme will be developed which is realized in a new optimization algorithm for the problem domain of molecular docking. This chapter will introduce several examples of self-organization that can be found in social insects, as well as current computational techniques which are based these self-organizing principles.

1.1. Collective behaviour in social insects

While decentralized collective behaviour can be observed across the whole animal kingdom (Sumpter, 2010), the collective behaviour observed in social insect societies is of particular interest for computer science. Social insect colonies contain reproductive entities (one or multiple queens) and a huge non-reproductive workforce. Worker individuals are very closely related (Wilson, 1971) and thus usually show the same behavioural repertoire. In addition social insects are “relatively” simple (in terms of cognitive abilities) and thus operate by means of relatively simple rules. However, in spite of their genetic homogeneity and simplicity, as a whole they are able to tackle a number of very difficult tasks.

In social insects division of labour, also known as *polyethism*, constitutes a self-organizing process. Due to its decentralized and adaptive nature it allows a colony to adjust its usually large workforce (i.e., thousands of individuals) according to its needs. As polyethism is very robust and flexible it has been proposed as one of the most important factors for the ecological success of social insects (Wilson, 2001).

The ways in which polyethism is achieved in social insects are as diverse as social insects themselves. Some ant species exhibit caste polyethism, meaning that the colony contains different morphological castes, which are specialized for different tasks (Wilson, 1971). One example of caste polyethism can be found in the ant species *Pheidole pallidula*. Colonies of this species contain two worker classes: minors and majors. These classes differ in terms of body and head size and take over different tasks in the colony. While minors work on tasks such as broodcare and foraging, majors function as soldiers of the colony and defend it from enemies (Detrain and Pasteels, 1992). Another form of polyethism that can be found in social insects is so called temporal polyethism, where individuals will specialize on certain tasks depending on their age. Honeybees are an example of social insects that exhibit temporal polyethism. While young workers can be usually found in the hive, taking over duties like brood and nest care, more senior individuals will guard the nest or forage (Huang and Robinson, 1996). A third form of polyethism found in social insects is so called genetic polyethism. Many insect species have been found to be polyandrous (their queens will mate with multiple males), which causes slight genetic diversity among the worker population. This genetic diversity has been linked to division of labour, as it can influence an individual's task preference (Robinson and Page, 1989). One example for genetic polyethism can be found in honeybees, where a worker's the sucrose receptiveness is determined by its patriline (Scheiner et al., 2004). Individuals that are highly receptive to sucrose will forage for water and nectar, while those that are less receptive will only forage for pollen.

Foraging constitutes one of the main duties in a social insect colony. It requires the proactive discovery of resources as well as the exploitation of known resources. In social insects different foraging and information-distribution mechanisms have evolved according to the species' environment. Many ant species use recruitment pheromone to guide fellow foragers towards potential food sources (Wilson, 1971). When returning from a candidate resource to the nest-site, an ant will deposit pheromones which form a trail leading others towards the resource. For the ant species *Lasius niger* it has been shown that trail recruitment can lead to a decision-making process when two food sources are discovered simultaneously with no prior established pheromone trail, which leads to the exploitation of the richer food source and an abandonment of the other source (Beckers et al., 1990). When presented with food sources which are of different distance to the nest-site, pheromone recruitment also enables a colony to decide on food sources located closest to the nest-site (Beckers et al., 1993). Using a double bridge experiment, where a food source can be reached on paths of multiple lengths, Goss et al. (1989) were able to show that the Argentine ant *Iridomyrmex humilis* will choose the shortest path towards a food source, by means of a stronger reinforcement of the shorter path due to faster travel times. In a recent study, Dussutour et al. (2009) were able to show that ants of the species *Pheidole megacephala* are able to constantly re-evaluate their decision and thus change their decision in dynamic environments where the quality of the food sources changes over time. Pheromone recruitment not only permits a colony to exploit and decide upon resources, it has also been shown that pheromone trails can evolve to transportation networks which resemble minimum spanning trees or Steiner networks (Latty et al., 2011).

Not all ants use pheromone for recruitment. Another strategy in ants is signal-based recruitment. After finding a potential resource an individual will return to the nest-site and signal that they found something, trying to gather followers which can then be led to the discovered resource. A special form of signal-based recruitment is tandem running, where an individual will lead a single recruit towards a resource via antennae contact (Franks and Richardson, 2006).

In honeybees other recruitment strategies have evolved. Upon returning to the nest from a discovered resource a honeybee will perform a dance on the so-called "dance floor", an area in the hive typically close to the entrance (Seeley, 2010). This dance, better known as the waggle dance (von Frisch, 1967), acts as a recruitment signal for idle bees. As well as the directions to the found resource, it also encodes its quality, which enables a quality-dependent recruitment rate and thus an optimal distribution of foragers on available resources with respect to their resource utility. While the waggle dance has been shown to be a very effective recruitment strategy in the case where resources are few and of poor or variable quality (Dornhaus et al., 2006; Grüter and Ratnieks, 2011) its importance in resource richer situations (e.g., flowering periods) is still debated (Grüter and Farina, 2009) (foraging in honeybees is described in more detail in Chapter 5).

Another well studied self-organizing aspect of social insect colonies is colony migration. There are several reasons why an insect colony might decide to abandon an established nest and search for a new home, such as seasonal changes, reproduction or the destruction of its current home. Before a colony can it first needs to locate a suitable location. If more than one option exists, it must select the best among those. This decentralized decision-making process is commonly referred to as "nest-site selection" and usually involves only a fraction of a colony's population (Visscher, 2007).

As homeless colonies are often exposed to the environment during the selection process, they face the additional challenge of deciding on the best nest-site as fast as possible. Social insects thus face a so called speed-accuracy trade-off. The selection process starts off with scouts exploring a colony's environment. Upon the discovery of a potential nest-site these scouts will report back to the colony in order to recruit other individuals to evaluate and campaign for the found nest-site. Recruitment techniques used in the nest-site selection process do not differ from those used during foraging (e.g., honeybees recruit using the waggle dance, some ant species use tandem running as a recruitment strategy). However, the main difference between nest-site selection and foraging is that during nest-site selection a colony needs to converge towards a single choice.

It has been shown that the nest-site selection process in ants and bees is able to tune itself with respect to the available options as well as environmental conditions (Franks et al., 2003; Passino and Seeley, 2006; Sumpter and Pratt, 2009). Franks et al. showed that the ant species *Leptothorax albipennis* will adapt its decision-making speed based on the current environmental conditions. In situations where the colony is exposed to a harsh environment, it will decide faster on a new nest-site in comparison to situations where this is not the case. Honeybees and ants have been shown to adapt their decision-making speed depending on the nesting options already under consideration (Passino and Seeley, 2006; Sumpter and Pratt, 2009). If all available options are of poor quality, this will lead to a decrease in recruitment speed, slowing down the decision-making process and prolonging the proactive search for alternative nest-sites. In contrast, the discovery of good nest-sites is usually accompanied by rapid recruitment towards the nest-site and thus speeds up the decision-making process.

As nest-site selection in social insects is decentralized, a decision on an option also has to be established in a decentralized manner. Social insects have evolved consensus decision making (Pratt et al., 2002; Seeley and Visscher, 2004). While evaluating potential nest-sites an individual will also take into account how many other individuals are evaluating this site. If the number of individuals reaches a critical level, which is referred to as quorum, a decision has been reached. Upon sensing this quorum the individuals at the site will return to their colony and start the migration process.

1.2. Swarm intelligence

The definition of "swarm intelligence" was coined by Beni and Wang (1989) in the context robotics as "systems of non-intelligent robots exhibiting collectively intelligent behaviour evident in the ability to unpredictably produce 'specific' ([i.e.] not in a statistical sense) ordered patterns of matter in the external environment" (Beni and Wang, 1989, p.2). Bonabeau et al. (1999) refined Beni and Wang's initial definition, by including "any attempt to design algorithms or distributed problem-solving devices inspired by the collective behaviour of social insect colonies and other animal societies" (Bonabeau et al., 1999, p.7).

Many insect-inspired techniques have been proposed in the context of swarm robotics. Mechanisms inspired by division of labour in ants as well as honeybees have been proposed for the control of division of labour in robot swarms (e.g., Krieger et al. (2000); Labella et al. (2006); Zhang et al. (2007)). The pheromone laying behaviour observed in ants has inspired navigational map and marker mechanisms in robotics (e.g., Russell (1997); Vaughan et al. (2002)). Several communication mechanisms proposed in swarm robotics are based on communication principles of social insects (e.g., Schmickl and Crailsheim (2008); Trianni and Dorigo (2006)). Interestingly, biology has recently begun to directly profit from advances made in robotics by using robots in behavioural experiments with animal collectives such as cockroaches and fish shoals, as they enable controlled experimental set-ups (Garnier, 2011; Halloy et al., 2007)

In the context of data clustering a number of swarm or ant-based clustering techniques have been proposed (Handl and Meyer, 2007). These techniques are based on stigmergic mechanisms found in social insects (Theraulaz and Bonabeau, 1999), where individual-based changes within the environment will exert feedback on future decisions of the individuals. Pheromone is an example of a stigmergic mechanism, where the deposit of pheromone will increase an individual's likelihood to reuse a given path and thus deposit even more pheromone on it. Another example of stigmergy which can be seen as one of the prime inspirations of ant-based clustering mechanisms are aggregation and sorting strategies in social insects. Individuals are more likely to drop items in areas where many items are already present, which leads to clusters of items. For data clustering, similar mechanisms can be used to achieve sorting of high-dimensional data.

Task allocation and foraging mechanisms found in social insects have not only been used in the context of robotics. Response threshold based task allocation mechanisms found in social insects have been suggested as control mechanisms for the workflow in factories (Bonabeau et al., 1997; Cicirello and Smith, 2004). Nakrani and Tovey (2004) introduced a mechanism inspired by honeybee foraging to regulate sever allocation. Several routing algorithms used in telecommunication networks are based on social insects. Di Caro and Dorigo (1998) introduced AntNet, a package routing algorithm which is based on pheromone trails and enables routing in dynamic networks. Wedde et al. (2004) introduced BeeHive, a package routing algorithm based on honeybee foraging. These algorithms were later extended for mobile ad-hoc networks (Di Caro et al., 2005; Wedde et al., 2005)

Social insects have also inspired the design of several population-based meta-heuristics for optimization. Ant colony optimization (ACO), introduced by Dorigo (1992), is based on the observation that some pheromone-laying ant species will converge towards the shortest path between nest and food source (Goss et al., 1989). Ant colony optimization uses virtual pheromones which denote the goodness of a solution in terms of fitness as a heuristic for solution construction. Over iterative construction steps, solution elements that are in solutions of good quality will be reinforced more strongly than those which are not, which leads to an increased selection of those elements in future construction steps. While ant colony optimization was originally designed for the Travelling Salesperson Problem, it is nowadays a very popular optimization technique that has been extended and used for various problem domains (for an overview see Dorigo and Stützle 2004)

Several other optimization algorithms are based on the behaviour of ants. Monmarché et al. (2000) proposed an algorithm based on the foraging behaviour of the ant species *Pachycondyla apicalis*. These ants hunt in close proximity of their nest. The proposed algorithm utilizes this behaviour by conducting parallel local searches around the current nest position in search space, followed by a nest relocation. Greenwood and Abbass

(2007) proposed a local search algorithm for continuous spaces based on army ant swarm raids. Similarly to army ants roaming the environment by the formation of tree-shaped trails (Franks et al., 1991), this algorithm is able to search a continuous fitness landscape.

Kennedy and Eberhart (1995) proposed particle swarm optimization (PSO), which draws its inspiration from group guidance in animal collectives. Here a swarm of particles is randomly placed in a multi-dimensional search space. During an optimization run these particles move through the search space. Their movement is influenced by "cognitive" information (best known position) as well as "social" information (best known position of particles in their neighbourhood; global and local variants exist). This simple behaviour will lead to a convergence towards an optimum in the search space. Like ACO, PSO is a very popular optimization technique that has been extended and used in many problem domains (for an overview see Poli et al. 2007).

Several algorithms based on the honeybees' collective behaviour have been developed and applied to various domains such as network routing, robotics, multi-agent systems, and optimization. Existing optimization algorithms based on principles of honeybee behaviour usually mimic either foraging or mating behaviour. Mating-inspired optimization algorithms are closely related to methods found in evolutionary computation. They are based on the fact that genetic heterogeneity among workers typically increases a colony's fitness (Fuchs and Schade, 1994). In honeybees genetic heterogeneity is achieved via the queen mating with several males (polyandry). While some mating inspired methods constitute new operators for existing methods in evolutionary computation (e.g., Jung (2003); Karci (2004); Sato and Hagiwara (1997)), others try to mimic the mating flight both on a behavioural and genetic level (Abbass, 2001a).

Optimization algorithms based on the foraging concept consist of a number of agents, known as artificial bees. As in nature, the purpose of the agents is twofold. On the one hand they search for new solutions (find food sources) in problem space, on the other hand they try to improve (exploit food sources) existing solutions using local search. The ratio between exploration and exploitation behaviour depends on the number and quality of available solutions. Several foraging-based algorithms have been proposed such as the artificial bee colony optimization (ABC) (Karaboga (2005)), the bees algorithm (BA) (Pham et al. (2006b)), the bee colony optimization (BCO) (Teodorovic and Dell'Orco, 2005) or the bee colony optimization algorithm (BCOA) (Chong et al., 2006). A detailed overview on bee-inspired algorithms is presented in Chapter 8.

1.3. Outline

The thesis covers several aspects of self-organization principles found in social insects. The first part of this thesis (Chapters 2, 3 and 4) will investigate division of labour on the basis of response threshold models. The second part of this thesis (Chapters 5, 6, 7, 8) studies the nest-site selection and guidance behaviour of honeybees. Based on nest-site selection behaviour, a new bee-inspired optimization algorithm is proposed and applied to the domain of molecular docking.

Chapter 2 introduces the family of response threshold models. Two model variants, the fixed response threshold model and the threshold reinforcement model, are presented formally. In addition, previous theoretical work which has utilized these models is outlined.

Originally published in Diwold et al. (2009a), Chapter 3 investigates division of labour in dynamic environments. On the basis of a threshold reinforcement model with fluctuating demands it is investigated to what extent different sized colonies are able to adapt to changing work loads. In addition, group-size dependent specialization over consecutive demand changes is investigated.

Chapter 4 introduces an extension of the threshold reinforcement model that allows the incorporation of spatial task distribution in such models. The aim of this Chapter is to investigate to what extent spatial task separation, which is often observed in social insect colonies, influences the productivity of a threshold system. In addition, several methods that can achieve such separation (i.e., via sorting tasks) are studied and compared regarding their adaptivity. This chapter has been previously published in Diwold et al. (2009b).

Chapter 5 gives an overview of the biological mechanisms underlying the self-organization in the European honeybee species *Apis mellifera*. Parts of this chapter were previously published in Diwold et al. (2011b).

Chapter 6 investigates to what extent the nest-site selection process of honeybees is shaped by the environment it operates in. To investigate this question a spatial nest-site selection model for honeybees is developed, which is then tested in different spatial set-ups.

Chapter 7 explores the guidance mechanisms that underlie swarm navigation during migration. First, the validity of two potential mechanisms is compared with respect to empirical data. In addition, the impact of directional dissent on guidance in honeybees is studied. Parts of this chapter were previously published in Diwold et al. (2011d).

Chapter 8 reviews current bee-inspired algorithms (previously published in Diwold et al. (2011b)) and introduces nest-site selection found in honeybees as a new optimization paradigm. First the optimization potential of the nest-site selection mechanism in noisy and dynamic environments is tested on the basis of the spatial nest-site selection model introduced in Chapter 6. In addition, an iterative application of nest-site selection on primitive search spaces as a means of function optimization is tested, previously published in Diwold et al. (2010). As the results are quite promising, a general optimization scheme is proposed along with an algorithm that is applied to the domain of molecular docking, which appeared in Diwold et al. (2011c).

Part I.

Response threshold models of division of labour in social insects

Worker polymorphism in the marauder ant *Pheidologeton affinis*. Photo by Alexander Wild, reproduced with kind permission[1].

Understanding the mechanisms underlying division of labour in social insects is not only important as a means of understanding the functionality of social insect societies. The behavioural mechanisms underlying division of labour are simple, scalable and robust, and have thus been previously used in the design of bio-inspired techniques in the field of optimization (Campos et al., 2000; Cicirello and Smith, 2004), multi-agent systems (Lemmens et al., 2008) and robotics (Krieger et al., 2000; Labella et al., 2006; Zhang et al., 2007).

As outlined in Chapter 1 there are several ways in which polyethism can be achieved in social insects (i.e., caste polyethism, temporal polyethism and genetic polyethism). None of these concepts is the exclusive mechanism underlying division of labour. Instead, social insect colonies usually exhibit an interplay between several forms of polyethism. This ensures the robustness of division of labour, as a colony can still function in the case of extreme events, if for example a majority of the colony dies (Wilson, 1984). Several models of polyethism in social insects have been proposed (for a general overview the interested reader should refer to the review of Beshers et al. 2001). In the following chapters response threshold models will be used to investigate division of labour. Response threshold models are based on the response threshold hypothesis, which assumes that an individual's action is a response to task-related stimuli which they perceive in their environment. Individuals are thought to have an intrinsic response threshold for each task,

---

[1] www.alexanderwild.com

which determines how sensitive they are to the task-specific stimulus. As outlined earlier, the forms of polyethism within social insects are diverse, thus a number of factors such as experience, genotype and physiological state can influence an individual's receptiveness to a task. If a stimulus is strong enough (i.e., exceeds a worker's response threshold), then the individual is very likely to start engaging in the task associated with the stimulus. Given an inhomogeneous distribution of internal threshold levels among individuals within a colony (BONABEAU ET AL., 1996; ROBINSON AND PAGE, 1989), the colony will be able to divide labour among its individuals. In the following we will investigate the behaviour of response threshold models under different environmental conditions. After a formal introduction to these models in Chapter 2, Chapter 3 studies the behaviour of threshold models in environments that exhibit fluctuations in task demands. Chapter 4 investigates the influence of spatial task separation on the performance of response threshold models.

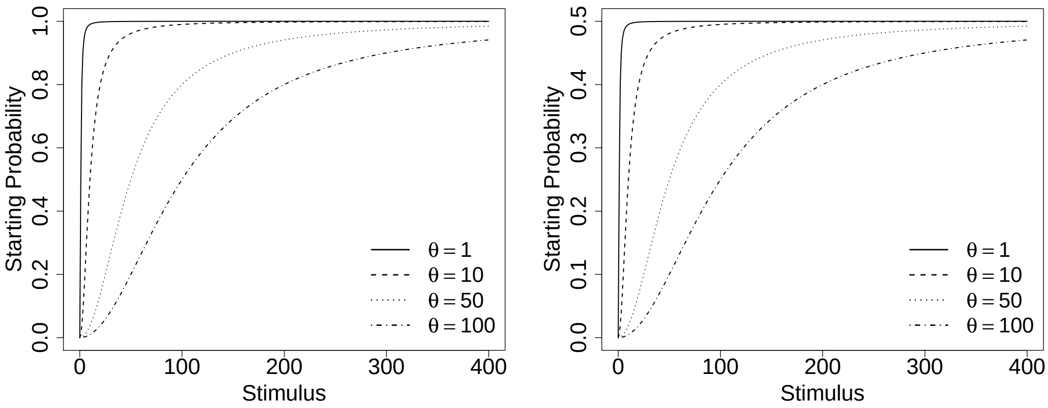
2. Response threshold models

Two types of models are generally used to capture the response threshold hypothesis in social insects: fixed response threshold models and threshold reinforcement models. Fixed response threshold models (FRMs) were formally introduced by Bonabeau et al. (1996) and assume that an individual's response threshold is fixed over its lifetime. FTMs are supported by a series of theoretical studies (Bonabeau et al., 1998; Jeanson et al., 2007; Waibel et al., 2006). In addition, several experimental studies have shown that the response threshold hypothesis can be used to account for the regulation of a number of tasks in social insects, such as removal of dead nest mates (Robinson and Page, 1989), thermoregulation (Jones et al., 2004), preferred foraging-task choice (Fewell and Bertram, 1999; Pankiw and Page, 1999; Scheiner et al., 2004) and defence (Detrain and Pasteels, 1992).

For some social insects, such as the ant species *Leptothorax* (Sendova-Franks and Franks, 1994) and honeybees (Ben-Shahar et al., 2000; Withers et al., 1993), it has been observed that individuals' thresholds are not necessarily fixed but change over time in a reinforcement-like manner. In order to account for this behaviour, the threshold reinforcement model, which constitutes a refinement of the basic FTM, was introduced by Theraulaz et al. (1998). In this model the thresholds of an individual change over time according to learning and forgetting rates. When an individual works on a given task its corresponding threshold will decrease, causing the individual to be more receptive to this task. Not working on a task however will increase an individual's threshold, thus making the individual more insensitive for the task. This can cause individuals to specialize (i.e., they have a low threshold) for a task. TRMs have been used to study which factors drive specialization in social insects (Gautrais et al., 2002; Merkle and Middendorf, 2004).

2.1. Fixed response threshold models

Typically an FTM consists of N individuals and m tasks $T_1, \dots, T_m$. Each task T_j is associated with a task-specific stimulus value $S_j \geq 0$. Accordingly, each individual i has a task-specific threshold value $\theta_{i,j}$ for which $0 \leq \theta_{i,j}$ holds.



(a) One Task

(b) Two Tasks

Figure 2.1.: Stimulus response curves for different threshold values θ in a threshold reinforcement model containing (a) one and (b) two tasks. Higher θ require higher stimuli in order to trigger a response.

Let X denote the state of an individual that determines the task the individual is currently working on. An individual has $m+1$ possible states: it can either work on one of the m tasks, or stay idle. Idle individuals that encounter a certain task T_j will start to work on that task with the probability

$$P(X_i = j|T_j) = \frac{{S_j}^2}{({S_j}^2 + {\theta_{ij}}^2)} \tag{2.1}$$

If the model contains more than one task ($m > 1$), a task distribution that determines the encounter likelihood of a task has to be appointed. Here it will be assumed that all tasks can be encountered with the same likelihood, thus the probability that an idle individual i will start working on task T_j in a model with m tasks is given by

$$P(X_i = j) = \frac{1}{m} \cdot \frac{{S_j}^2}{({S_j}^2 + {\theta_{ij}}^2)} \tag{2.2}$$

Figure 2.1 illustrates how an individual's threshold for a task affects the probability of the individual to work on that task in a response threshold model under one (Figure 2.1(a)) and two tasks (Figure 2.1(b)). As can be seen, the level of an individual's threshold for a task negatively correlates with its probability of starting to work on the task given a certain task stimulus level. In addition the number of tasks present in the system will determine the maximum probability of engaging in a task under the assumption that tasks are equally encountered.

Each time step an individual is engaged in a task, it will do α units of work on this task ($\alpha > 0$). With a certain probability of $p \geq 0$, individuals that are active at time step t will become idle in the next time step $t+1$. An individual that turns idle will remain idle for at least one time step before it can re-engage in a task.

In the following we determine the maximal amount of work that a colony with N individuals and m tasks can perform on average in one time step per task, denoted by $W_{\max}$ (see also GAUTRAIS ET AL. 2002). For this we assume a system where all individuals will be engaged in a task immediately after one idle time step. Let p_{max} be the fraction of individuals that are working when the system with maximally working individuals is in an equilibrium state. Then $p_{max} \cdot p$ denotes the fraction of individuals that become idle at each time step, which has to be equal to the fraction of idle individuals that become engaged $(1 - p_{max})$. This equation leads to a fraction of $p_{max} = 1/(1+p)$ individuals that are not idle. As the workload is distributed by N individuals on m tasks, and as α units of work are done by each individual in a time step

$$W_{\max} = \frac{N}{m} \cdot \frac{1}{1+p} \cdot \alpha \tag{2.3}$$

To establish the task stimuli each task T_j is associated with a demand parameter $0 \leq D_j \leq 1$. This parameter denotes the fraction of work a task requires per time step to keep its stimulus level constant. As $W_{\max}$ denotes the maximal amount of work that can on average be performed on a task per time step, a task T_j with demand D_j will require

$$\delta_j := D_j \cdot W_{max} \tag{2.4}$$

work per time step. If for example $D_j = 1$, a colony is required to perform $1 \cdot W_{max}$ amount of work on task T_j per time step to maintain the stimulus value S_j at the same level. If a colony is not able to maintain sufficient workforce for a task T_j, this will lead to an increase of the task-related stimulus S_j. If on the other hand the colony exceeds the amount of work necessary for the task, the stimulus will decrease. Formally, in each time step, the task-specific stimulus values are changed for each task T_j according to

$$S_j = S_j + \delta_j - E_j \cdot \alpha \tag{2.5}$$

where E_j is the number of individuals currently engaged in task T_j, and δ_j determines the additive value for the stimulus in each time step.

2.2. Threshold reinforcement models

Threshold reinforcement models differ from fixed response threshold models, as not only the task-related stimuli can change over time, but also the individuals' thresholds are able to change, denoting a change in individual task preference. In order to prevent thresholds from getting infinitely large, which would result in an individual never working on that task again, such models require maximal task thresholds. For each individual i, $\theta_{i,j} \leq \theta_j^{\max}$, where $\theta_j^{\max} \geq 0$ denotes the maximal threshold for task T_j.

At each time step, the threshold values for each task T_j are also updated for each individual i as follows:

- if i works on T_j then $\theta_{i,j} = max\{\theta_{i,j} - \xi, 0\}$
- if i did not work on T_j then $\theta_{i,j} = min\{\theta_{i,j} + \phi, {\theta_j}^{\max}\}$

Individuals are usually initialized with a threshold of 0.0 for each task. ξ is the learning parameter and ϕ the forgetting parameter. Learning and forgetting can occur at different speeds. Ben-Shahar et al. (2000) have shown that bees show their maximal response towards a specific odour after 5 to 6 learning steps. The strength of their response is reset to a low value after around 5 contacts with a different odour. In previous studies on threshold reinforcement models, learning and forgetting parameters of $\phi = 3.5$ and $\xi = 4.0$ were used (Gautrais et al., 2002; Merkle and Middendorf, 2004). Further discussion on the influence of learning and forgetting rates can be found in Gautrais et al. 2002.

Revision of the threshold reinforcement model Gautrais et al. (2002) used a TRM to study colony-size-dependent specialization. They concluded that increased colony size triggers specialization in social insect colonies. The model was re-examined and extended in Merkle and Middendorf (2004). It was noticed that stimuli-growth is colony dependent, as δ_j is proportional to the colony size (due to W_{max}, see Eq. 2.4). Stimuli will thus grow much more slowly in small colonies than in larger colonies (see Eq. 2.5). Individuals threshold values, on the other hand, are independent of colony size, which can lead to complications in the initial phase of the model. Like thresholds stimuli are usually initialized to 0. As stimuli values grow slowly in colonies of small size, individuals in such colonies are very unlikely to work in the first steps of the simulations. As a consequence their thresholds will increase, which makes it even more unlikely they will work in the next steps. If thresholds grow faster than stimuli this can lead to a situation where most individuals have task thresholds around $\theta_j^{\max}$ for each task. Such individuals exhibit similar activity for both tasks, which makes it nearly impossible for an individual to become a specialist.

In large colonies in which stimuli grow faster in the first time steps after initialization, this prohibits the above effect from taking place. An individual in a large colony will therefore start to work early on a task, hence its threshold for the task it is working on will stay near 0 and the threshold for the other tasks will increase by ξ each time step. This leads to individuals specializing in single tasks.

To remove this effect of colony size on specialization in the initial steps of the simulation Merkle et al. Merkle and Middendorf (2004) suggested a demand variation phase after the initialization, which is an alternation of long periods of low and high demands for the tasks in the system. After such a demand variation phase the system does not depend any more on the initial conditions which caused the artefact described above.

Another criticism of the standard TRM raised by Merkle and Middendorf (2004) was that results gained with such a model fundamentally depend on simulation time. This is due to the fact that the differentiation of individuals with respect to specialization or activity level is possible only over finite time periods. As an individual's probability to change from its current to any other possible state is non-zero, there is no difference in behaviour over infinite time (i.e., all individuals would have the same degree of specialization and activity). As individuals in previous TRMs had an infinite life span, all observed differentiations depended on the simulation time. In order to overcome this problem, a finite life span was incorporated into the model by introducing a maximum age $a_{\max}$. The individuals are initialized with an age that is chosen uniformly at random between 0 and $a_{\max}$. When the age of an individual reaches $a_{\max}$ the individual is reset. This means that it is removed from any task it is currently working on and its task thresholds and age are set to 0. For a better understanding, the algorithm underlying a simulation with a revised TRM is outlined in Algorithm 1.

2.3. Related work: Theoretical studies on response threshold models

Several theoretical studies have used FTMs as well as TRMs to study various aspects of division of labour in insect societies. Bonabeau et al. (1996) provided the first formal description of FTMs. Prior to the formal introduction in 1996, the model was used informally in several studies (e.g., Robinson 1987, 1992. In their article, Bonabeau et al. showed that threshold models are indeed able to maintain division of labour in social insects. Further they were able to reproduce empirical task-response characteristics of the ant genus *Pheidole* reported by Wilson (1984). The ant genus *Pheidole* features two morphological castes, minors and majors, which differ in their task preference (majors function as guards, while minors are engaged in foraging and nest-care tasks). Wilson

Algorithm 1 Threshold Reinforcement Model

```
initialize model parameters
reset individuals
for all simulation steps do
    for all individuals do
        if working on task then
            stop working with probability p;
        else
            if not active previous step then
                for all tasks do
                    choose to work on task according to probability $P(X_i)$
                end for
            end if
        end if
        age ++
    end for
    for all tasks do
        update task stimuli
    end for
    for all individuals do
        for all tasks do
            if engaged in task then
                decrease individuals threshold for task
            else
                increase individuals threshold for task
            end if
        end for
        if age == maximal age then
            reset age to zero
            reset thresholds
        end if
    end for
end for
```

noted that these roles are not necessary fixed. Upon a decline in the number of minors present in a colony, majors will start to take over tasks which they would normally not work on. Bonabeau et al. were able to show that such task-preference dynamics can be achieved in an FTM with two distinct castes that have differing thresholds for certain jobs, thus allowing for a robust and fault tolerant mechanism for division of labour.

Theraulaz et al. (1998) introduced the first model of threshold reinforcement, which allows individuals to change their thresholds via learning and forgetting parameters. They were able to show that such a reinforcement process can lead to the emergence of specialization within an insect colony which initially contains identical individuals. Threshold reinforcement can thus account for task allocation and within-caste specialization in social insects. In addition Theraulaz et al. also studied colony response towards perturbations (the removal of specialists from the colony) in a threshold reinforcement set-up.

Gautrais et al. (2002) used a TRM to study colony-size dependency of specialization. As Theraulaz et al. (1998) pointed out, the reinforcement of threshold can lead to task specialization within social insect colonies. In order to test the effect of colony size on the level of specialization within a colony, the model behaviour was studied under several colony sizes and task demand concentrations. Gautrais et al.'s findings showed that individuals in large colonies tend to differentiate in terms of activity patterns, which leads to task specialization, while this is not the case for small colonies. Demand was identified as an additional factor in fostering specialization in social insects. Gautrais et al. results are in agreement with previous empirical studies, which found that organization and task-specialization within certain social insect species such as wasps are colony-size dependent (Jeanne and Nordheim, 1996; Karsai and Wenzel, 2000).

As pointed out above Merkle and Middendorf (2004) re-examined and extended Gautrais et al. (2002)'s TRM. They suggest the use of an initialization phase at the beginning of a simulation to remove effects of colony-size dependent stimuli growth which influence the specialization within a model at early stages. In addition they suggested the introduction of a finite life-time to detach simulation results from the model runtime. In addition they studied the effects of age-dependent thresholds as well as task competition on specialization in social insects. Merkle and Middendorf (2004) were not able to report colony-size dependent specialization under normal conditions. However, competition for work (i.e., only a fraction of individuals is allowed to work on a given task) can lead to colony-size dependent specialization, depending on the strength of the competition.

For successful and robust division of labour a colony needs an inhomogeneous distribution of internal task-threshold levels among individuals. Genetic variation among individuals has been identified as one of the main factors causing such inhomogeneous task preferences in a population (Robinson and Page, 1989). Waibel et al. (2006) further studied this aspect, by investigating what effect the mapping between phenotype and genotype in terms of individual task thresholds has on division of labour. Using artificial evolution Waibel et al. compared three genetic architectures "deterministic mapping", "probabilistic mapping" and "dynamic mapping". Under deterministic mapping an individual's task thresholds were strictly determined by its genotype. Under probabilistic mapping the genotype impacted an individuals probability to engage in a task. In the case of dynamic mapping an individuals' task preference not only depended on its own genotype but also on the behaviour of other colony members. Waibel et al.'s findings suggest that in order to create a robust process of division of labour that is able to deal with perturbations, strict or probabilistic mappings are not sufficient. In order for the colony to be able to react to perturbations its individuals need to exhibit behavioural flexibility, which can be achieved for example via dynamic mapping.

In contrast to Gautrais et al. (2002) and Merkle and Middendorf (2004), Jeanson et al. (2007) studied specialization in social insects using FTMs. Jeanson et al. found that even in models that lack reinforcement, increased division of labour is positively correlated with group-size. However, division of labour also depends on other factors such as number of tasks and the task demands, which are group-size dependent. Their results suggest that low demand and high task number facilitate an increase in division of labour. In nature an increase in colony size is usually accompanied by an increase in task-number. For example foraging in fungus garden ants only sets in when a certain colony size is reached (Fernandez-Marin et al., 2003). In contrast, demand will decrease with increased colony size as the number of tasks only scales with colony size up to a certain point – the number of tasks is finite and will thus saturate at some point, while the colony's size can still increase further. This interdependence between number of tasks, demand and colony-size could explain why division of labour is often found to be correlated with colony size.

In a very recent study Richardson et al. (2011) introduced the so-called spatial fixed-threshold model (SFTM). In contrast to the standard threshold models the SFTM operates on a one or two-dimensional lattice. Each cell of the lattice corresponds to a potential task and can exhibit a certain stimulus which can increase over time. Agents with fixed thresholds populate the lattice and will work on the task located in their current cell if the associated task stimulus exceeds their threshold. By working on a task an individual will decrease the stimulus to a level that is unrecognisable to the individual (i.e., lower than its threshold). Individuals are mobile and can move onto neighbouring cells if they are not occupied. Using this simple model Richardson et al. were able to show that given a heterogeneous set of individuals and stimuli non-random spatial structures will arise both on the individual as well as on the stimulus level, thus leading to homœostasis.

3. Division of labour in dynamic environments

In their natural habitats, social insect colonies are embedded in dynamic environments. Conditions in such environments can change on an hourly, daily and seasonal basis. While some changes in the environment will leave a colony unaffected, others will impact the colony and can cause a switch in task necessities. In order for a colony to survive, it must be able to react appropriately to such changes by adapting its work-forces to the change in demand.

A number of events such as a massive loss of workers from a specific task group (Wilson, 1984), migration (Langridge et al., 2008), increase in temperature (Jones et al., 2004) or loss of reserves (Fewell and Bertram, 1999) can occur in social insect colonies, which require a rapid readjustment of the work-force. As the survival and well-being of a social insect colony crucially depends on a colony's ability to react to environmental changes, it is important to investigate this particular aspect in response threshold models.

Langridge et al. (2008) investigated the behaviour of a *Temnothorax albipennis* ant colony over several consecutive migrations. Their findings show that repeated migrations lead to a specialization within the colony. While the transportation during a colony's migration is performed by many individuals during the first migrations, specialists for this task develop after a few migrations. These specialists dominate the transportation in later migrations and increase the colony's overall performance during those migrations.

This suggests that certain events in the environment can actively drive specialization in social insects. Since a previous theoretical study (Merkle and Middendorf, 2004) has shown specialization is not innate to threshold models per se, it is interesting to investigate whether consecutive "migration-like processes" in response threshold models can trigger specialization.

Here TRMs which incorporate dynamic environments with varying demand for work will be used to study the influence of rapid environmental changes on division of labour. This work represents a continuation of Diwold (2005), where response threshold models with dynamic environments were first introduced. In the course of this chapter we will see that dynamic environments indeed impact a colony's performance. Depending on the size of the colony such changes can cause overworking or underworking for the task that changed in demand (i.e., working more or less than the ideal amount). By adjusting the number of possible learning steps, which correspond to changes in the maximal threshold values relative to a colony's size, the performance of colonies in dynamic environments can be improved. In addition, a setup inspired by repeated migration behaviour (LANGRIDGE ET AL., 2008) is investigated. It is shown that colony-size dependent learning rates will affect a colony's ability to maintain an activity onset for a reappearing task and can thus be seen as a trigger for colony-size dependent specialization in social insect colonies.

3.1. Threshold reinforcement model in dynamic environments

To study the effect of dynamic environments on the response threshold concept, a two task TRM was used (a formal description of TRMs was provided in the last Chapter). Environmental changes can be incorporated into such models in the form of changes in the demand values during simulation time. This simplification is possible as we are only interested in environmental changes which affect task necessity. Given a standard TRM the demand value D_j represents the necessity of a task T_j for the colony. When increasing the demand of a certain task this will lead to an increase in work-force needed for the task in order to to keep the tasks stimulus S_j constant.

The threshold models used in previous studies used demands for the tasks that were fixed for the entire simulation time. One exception is found in BONABEAU ET AL. (1998), where the demand was doubled after some time steps to see how this affects the caste in the model. Apart from this single work increase, the effect of demand changes was not investigated further.

3.1.1. Adapting to dynamic environments

The first question, which is of particular interest in the case of dynamic environments, is how fast and how well different-sized colonies can adapt to dynamic environments. To introduce dynamic environments with changing demands over short time periods into the TRM, we first adopt an initialization phase, as described in Chapter 2. After this phase the demand variation starts at a time step t^a. This is done as follows. The simulation between time steps t^a and the end of the simulation at time step t^b is divided into equally long subintervals with length t^Δ. In each of the $(t^b - t^a)/t^\Delta$ intervals, a demand variation occurs for a certain number of time steps (details will be defined later). For clarity we will refer to such an interval as "demand variation interval" throughout this chapter. There are many possibilities for the implementation of a demand variation in a threshold reinforcement model. A demand variation could for example only affect one task at a time, or multiple tasks at the same or different times in the demand variation interval.

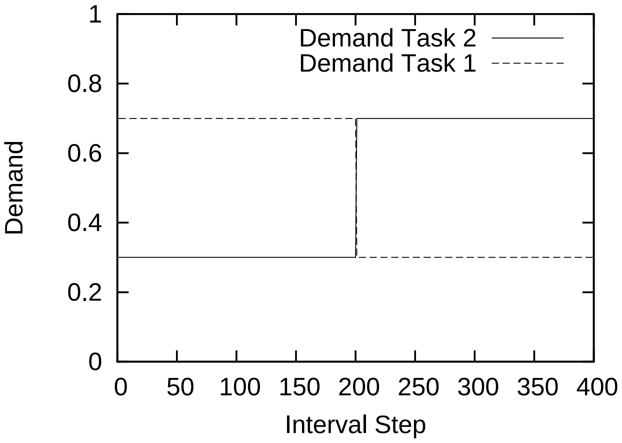


Figure 3.1.: Oscillation-like demand variation interval ($t^{\Delta} = 400$): standard demand values are $D_1^s = D_2^s = 0.3$, at complementary periods of the interval, the demand of one task changes to $D_1^c = D_2^c = 0.7$.

Here an oscillation-like demand variation will be studied, where the demands of both tasks are changed complementarity during each demand variation interval. Let D_j^s denote the standard demand value of a given task T_j and D_j^c the demand value of the task during an environmentally induced demand change. In the first $t^{\Delta}/2$ time steps of a demand variation interval the demand of task T_1 is changed to D_1^c while the demand of task T_2 remains at its standard level D_2^s. For the remaining $t^{\Delta}/2$ time steps the demand of task T_1 is reset to D_1^c while the demand of task T_2 is set to D_2^s. A demand variation interval with $t^{\Delta} = 400$ and standard demand values $D_1^s = D_2^s = 0.3$, that are changed to $D_1^c = D_2^c = 0.7$ during the task dependent period of the demand variation interval, is depicted in Figure 3.1.

3.1.2. Specialization in dynamic environments

As pointed out above LANGRIDGE ET AL.'s study on consecutive migrations in the ant species *Temnothorax albipennis* suggests that consecutive migrations trigger individual specialization. It is thus interesting to investigate whether a migration-like process can also lead to specialization in response threshold models.

In contrast to an oscillation-like demand variation, a migration-like process is established in the two-task threshold response model as follows. One of the tasks T_1 constitutes the modelled migration task. During non-migration times its demand D_1 is low but present. This is due to the fact that migration related actions are also required to some extent during times of no migration. Brood carrying and sorting behaviour are examples of these actions. Over the course of a migration such actions gain necessity, thus D_1 is increased. The other task T_2 represents the actions which are necessary at any time in the colony,

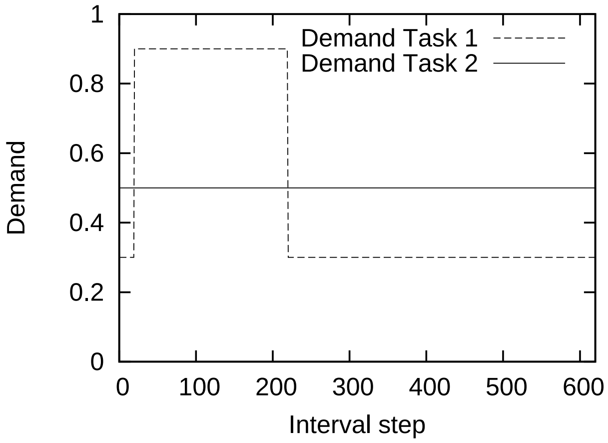


Figure 3.2.: Example migration interval with $t^{\Delta} = 600$, migration length $t^m = 200$, migration task T_1, other task T_2, demand of migration task during migration is $D_1 = D_1^c = 0.9$ and otherwise is $D_1 = D_1^s = 0.3$; demand of other task $D_2 = D_2^s = 0.5$. The migration is characterized by an increase of the demand related to the migration task. After migration the corresponding task demand is reset to D_1^c. The demand which is not related to the migration task is held constant.

regardless of whether a migration takes place or not. Thus, the demand associated with this task is not altered at any time.

In the simulation this is realized as follows. As in the setup outlined in the last section, the steps between time step t^a and the end of the simulation t^b are divided into equally long demand variation intervals of length t^{Δ}. In each of these intervals, a migration occurs for t^m steps ($t^m \leq t^{\Delta}$), this is referred to as migration interval (or just migration) of length t^m. During a migration the demand value of the migration task T_1 is set to a high value D_1^c. After the migration it is reset to its lower stationary value D_1^s. As already mentioned, the demand value D_2 of the other task T_2 is not altered at any point in the simulation and will remain at D_2^s during the whole simulation. The time span between the end of an migration interval and the start of the next migration interval is referred to as inter-migration interval.

The demand evolution of a typical migration within a demand variation interval length of $t^{\Delta} = 600$, $t^m = 200$ is depicted in Figure 3.2. In this example the demand for task T_1, D_1 is set to $D_1^c = 0.9$ during migration, otherwise it corresponds to $D_1^s = 0.3$, while the demand for task 2 D_2 is kept constant at $D_2^s = 0.5$. Please note that for clarity the migration interval starts at time step 20 in Figure 3.2.

3.2. Experimental setup

In order to study the influence of dynamic environments on polyethism in response threshold models a two task TRM model as described in Chapter 2 was used. Following Merkle and Middendorf (2004), in the first $t^{init} = 10000$ simulation steps an initialization-phase was performed to ensure that no artefacts due to initial conditions occur. An oscillation-like demand variation (Section 3.3.1 and 3.3.2) or migration (Section 3.3.3) started at step $t^a = 5 * t^{init}$ and lasted until the end of the simulation t^b. The individuals had a maximum age $a_{\max} = 1000$, the learning/forgetting parameters used were $\xi = 4.0$ and $\phi = 3.5$ (which corresponds to values previously used in the context of threshold reinforcement model (Gautrais et al., 2002; Merkle and Middendorf, 2004), α was set to 0.1 and $p = 0.2$. Other parameters such as colony size N, maximal threshold $\theta_{\max}$ as well as demand/migration interval length and demands differed among the experiments and are thus outlined in the respective subsection. All probabilities used in our simulations were derived from a uniform random distribution.

3.3. Experiments

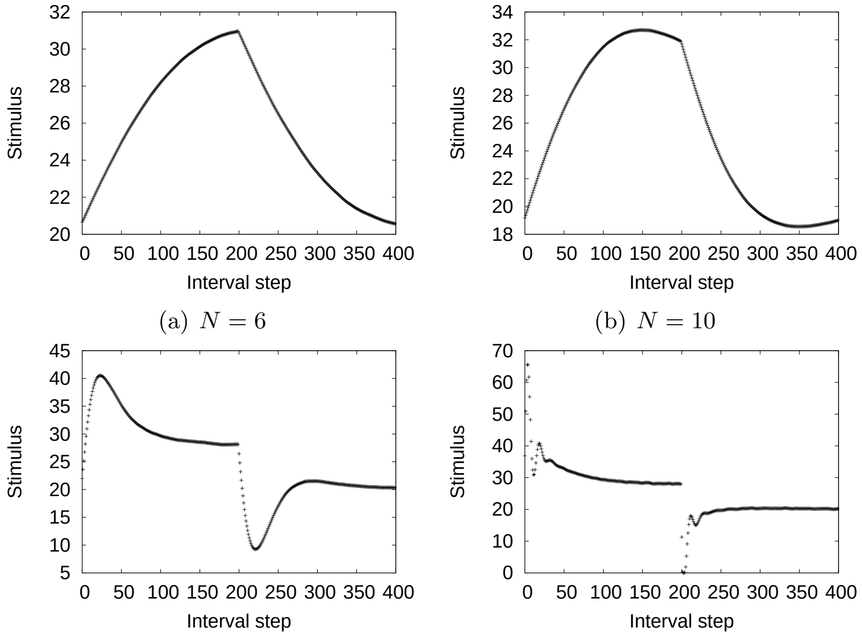
The following section is organized as follows: First the adaptation abilities of different-sized colonies under equal parameter conditions in dynamic environments are outlined. Then a way to increase a colony's adaptivity in such environments is presented. Finally the activity of different-sized colonies performing an iterative migration-like task is analysed.

3.3.1. Adapting to changing environments

Unless stated otherwise, the presented results were obtained from studies of the behaviour of different-sized colonies $N \in \{6, 10, 100, 1000\}$ using a maximal threshold $\theta_{\max} = 100$ in an environment exhibiting oscillation-like demand variation with $t^{\Delta} = 400$. Variation of demands is between low values a standard value of $D_1^s = D_2^s = 0.3$ and demand change value of $D_1^c = D_2^c = 0.7$ (as in Fig. 3.1). A simulation run lasted for 770000 simulation steps leading to 1800 demand variations per simulation.

These experiments were conducted to test how fast a colony is able to adapt to task-related changes in the environment. In order to investigate how the behaviour of different-sized colonies differs when confronted with environmental changes, colonies of different sizes were studied using the same set of parameters.

A good indicator of how fast a colony can adapt to a change in the environment are the task-related stimuli. A colony that has successfully adapted to environmental changes should be able to fulfil the colony's needs. This means that its individuals should neither work too little nor too much for the tasks present in the system. Such a behaviour should



(a) $N = 6$ (b) $N = 10$

(c) $N = 100$ (d) $N = 1000$

Figure 3.3.: Change of the stimulus associated with task T_1 in different-sized colonies during a whole oscillation-like demand variation interval $t^{\Delta} = 400$.

result in stable stimuli. On the other hand, stimuli should change if the colony has not yet adapted, depending on whether too much or too little work is being done on the given tasks.

Figure 3.3 depicts the evolution of the average stimuli values of task T_1 for colonies of size 6,10,100, and 1000. Remember that for task T_1 the demand is $D_1 = 0.7$ (while $D_2 = 0.3$) in the first 200 steps and $D_1 = 0.3$ (while $D_2 = 0.7$) in the remaining steps. As can be seen in Figure 3.3, stimulus behaviour differs strongly between different-sized colonies. While large colonies ($N = \{100, 1000\}$) are able to adapt fast to the change in task-demand, this does not hold for colonies with few individuals. Adaptation speed seems to scale with colony-size, meaning the larger the colony the faster the adaptation to changes in task demand.

How well a colony is able to adapt to environmental changes should also be reflected in the activity of the colony during a demand variation interval. The expected ideal amount of work W_j^{ideal} which should be done by a colony for a given task T_j (having a demand

Table 3.1.: Work statistics for the first $t^{\Delta}/2$ steps in an oscillation-like demand variation interval $t^{\Delta} = 400$; N: colony size, W_j^{ideal}: expected ideal amount of work for task T_j, W_j^{ave}: work for task T_j in the first 200 steps of a demand variation phase (averaged over all demand variation intervals); standard deviation is given in parentheses; $r_j := W_j^{ave}/W_j^{ideal}$: relative amount of ideal work done.

N	W_1^{ideal}	W_1^{ave}	r_1	W_2^{ideal}	W_2^{ave}	r_2
6	35.0	24.7 (6.7)	70.6%	15.0	25.3 (5.6)	168.7%
10	58.3	45.6 (6.6)	78.2%	25.0	37.9 (6.4)	151.6%
100	583.3	577.2 (4.7)	99.0%	250.0	257.9 (4.8)	103.2%
1000	5833.3	5842.6 (3.6)	100.1%	2500.0	2524.0 (6.1)	101.0%

parameter D_j) in t^v time steps can be easily calculated via (for details of this formula please refer to Chapter 2).

$$W_j^{ideal} = t^v \cdot \delta_j = t^v \cdot D_j \cdot \frac{N}{m} \cdot \frac{\alpha}{1+p} \tag{3.1}$$

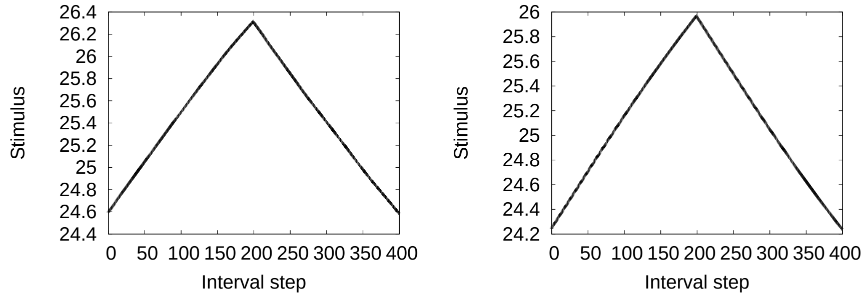
Here, we are interested in $t^v = 200$, which corresponds to half of the demand variation interval t^{Δ}, whether we look at the first of second half of t^{Δ} is irrelevant, as the task demands are symmetric.

Table 3.1 contains the amount of work, the expected ideal amount of work and the fraction of the ideal work that has been done for both tasks and colonies of sizes 6, 10, 100, and 1000, each calculated for the first 200 steps in a demand variation interval (i.e., $D_1 = 0.7$, $D_2 = 0.3$). All presented values are averaged over all demand variation intervals in one simulation run.

From the table it is clear that colonies of all sizes fulfil or exceed the ideal amount of work in the demand variation interval, for the task with the unchanged demand (i.e., task T_2). However, small colonies exceed the ideal amount of work far more than large colonies (e.g., the colony of size 6 works around 68.7% more than necessary while a colony of size 1000 exceeds the ideal amount of work only around 1%).

The ideal amount of work for the task with the increased demand is only fulfilled / exceeded by the largest colony (i.e., $N = 1000$). Smaller colonies are not able to accomplish the necessary workload. Furthermore, the ability to deal with a demand increase seems to be positively correlated with colony size.

Our suggestion is that the phenomenon seen here (i.e., the difference in the work performance during an demand variation interval) is due to the colony-size dependent stimuli. Stimuli trigger the awareness of a colony's individuals for a task. While stimuli in large colonies are very flexible (i.e., their level can increase or decrease significantly in one time



(a) $N = 6$ (b) $N = 1000$

Figure 3.4.: Change of the stimulus associated with task 1 in a model with colony-size independent stimuli, for colonies of size 6 and 1000 during a whole oscillation-like demand variation interval $t^{\Delta} = 400$.

step, see Eq. 2.5 in Chapter 2 for more detail), this does not hold for small colonies. This contrasts with the individual's threshold update – all individuals, no matter what colony size, exhibit the same threshold learning and forgetting rate. It seems therefore that individuals in small colonies are simply not as quickly aware of the workload dimensions they have to deal with, unlike the individuals in large colonies.

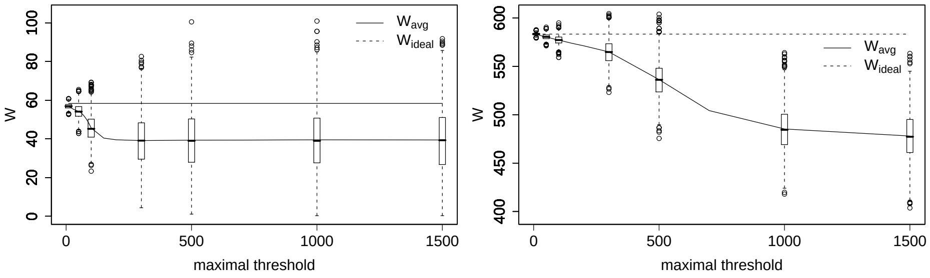
To check whether the observed differences between different-sized colonies in the amount of work, which is done during a demand variation interval, is a consequence of the colony-size dependency of the stimuli, we studied a modified model which uses colony size independent stimuli.

Within the model, colony-size independent stimuli can be achieved easily by the following modification of the stimulus update formula:

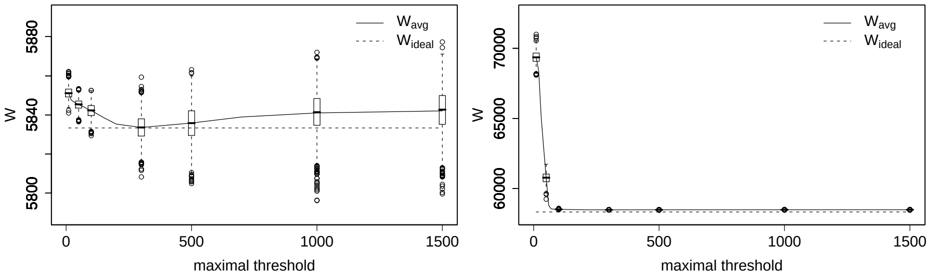
$$S_j = S_j + (\delta_j - E_j \cdot \alpha) \cdot 1/N \tag{3.2}$$

Table 3.2.: Work statistics for the first $t^{\Delta}/2$ steps in an oscillation-like demand variation interval $t^{\Delta} = 400$ in a model with colony-size independent stimuli update; for parameters see Table 3.1.

N	W_1^{ideal}	W_1^{ave}	r_1	W_2^{ideal}	W_2^{ave}	r_2
6	35.0	24.7 (8.9)	70.6%	15.0	25.4 (9.4)	169.3%
10	58.3	41.3 (12.9)	70.8%	25.0	42.0 (11.5)	168.0%
100	583.3	410.4 (39.6)	70.4%	250.0	422.6 (38.7)	169.0%
1000	5833.3	4105.7 (161.0)	70.6%	2500.0	4230.8 (190.8)	169.2%



(a) $N = 10$ (b) $N = 100$



(c) $N = 1000$ (d) $N = 10000$

Figure 3.5.: Difference of amount of work done by a colony of size N and the expected ideal amount of work W_j^{ideal}; results are depicted for task $j = 1$, colony sizes $N \in \{10, 100, 1000, 10000\}$ and maximal threshold values $\theta_{\max} \in \{10, 50, 100, 300, 500, 1000, 1500\}$; boxplots show the difference between the amount if work done and the expected ideal amount of work W_j^{ideal} to be done for all demand variation phases in all test runs; dotted line in the subfigures shows the value W_1^{ideal}.

This modification ensures that the stimuli have a growth rate that is independent of the colony size. Stimulus development and the work statistics of different-sized colonies in the modified model confirm our hypothesis. The stimulus evolution of colonies of size 6 and 1000 are depicted in Figure 3.4 and the work statistics for $N \in \{6, 10, 100, 1000\}$ are presented in Table 3.2. Within the modified model, the stimulus development and work statistics of different-sized colonies are relatively similar, which is reflected in the similar values of r_j for task $j \in \{1, 2\}$.

3.3.2. Increasing adaptivity in dynamic environments

As we have seen in the previous section, task adaptivity of TRMs in dynamic environments depends on colony size. While large colonies are able to adapt fast to environmental changes, small colonies lack these abilities.

In their study, Merkle and Middendorf (2004) pointed out that the maximal threshold $\theta_{\max}$ used in a TRM has a significant impact on the flexibility of the individuals when learning and forgetting rates are kept constant. There are two reasons for this. On the one hand, $\theta_{\max}$ has an impact on the magnitude of the stimuli in the system. Given a feasible demand for a task (i.e., one that the colony can fulfil) the stimuli will not rise much higher than $\theta_{\max}$. If a stimulus exceeds $\theta_{\max}$ this will cause a high percentage of the individuals to work on the corresponding task and will thus lead to a decrease in the stimulus. The other reason why $\theta_{\max}$ has an impact on the flexibility of the system lies in the fact that $\theta_{\max}$ determines how many learning steps are needed to switch between the two threshold extremes $\theta = 0$ and $\theta = \theta_{\max}$. Note that a large value for $\theta_{\max}$ corresponds to a large number of possible learning steps.

Using a small maximal threshold should result in a system with very flexible individuals, while larger maximal thresholds should have the opposite effect. To investigate whether different-sized colonies could benefit from different maximal thresholds, colonies of different sizes were studied in the system using maximal thresholds $\theta_{\max} \in \{10, 50, 100, 300, 500, 1000, 1500\}$ and a colony-size dependent stimulus update. Note that for $\xi = 4$ a small value $\theta_{\max} = 10$ corresponds to 2.5 learning steps and a high value $\theta_{\max} = 1500$ corresponds to 375 learning steps needed to reduce a response threshold θ from the maximal value to zero.

The work performance of each colony was monitored for each task in both demand variation interval halves. Due to the oscillation-like nature of the demand variation, task T_2 will behave like task T_1 in the second half of the demand variation interval and vice versa, thus only results for T_1 are presented. Results for colony sizes $N \in \{10, 100, 1000, 10000\}$ for task T_1 in the first half of the demand variation interval are depicted in Figure 3.5. For each combination of N and $\theta_{\max}$ a boxplot of the work done is given. The dotted line in the subfigures corresponds to the ideal amount of work W_1^{ideal} that should be done. Small colonies ($N \lesssim 100$) tend to adapt slowly to demand changes (take longer to begin working on the task with increased demand) if the maximal threshold is too large (see Figures 3.5(a) and 3.5(b)). Performance improves significantly when smaller maximal threshold values are used, as can be seen by the small difference between the amount of work done and the value of W_j^{ideal}. In large colonies $N \gtrsim 500$, a large maximal threshold does not corrupt a colony's ability to adapt to demand changes (see Figures 3.5(c) and 3.5(d)). This is due to the fact that the stimulus values can change more rapidly, as outlined in the last section. However, large colonies tend to overwork for the task.

Smaller colony sizes ($N \leq 10$) perform best with low $\theta_{\max}$, here 10. With increased colony size the best performance seems to shift proportionally to $\theta_{\max}$ – see Figures 3.5(b), 3.5(c) and 3.5(d). It should also be noted that very small values of very large colonies ($N = 10000$) combined with very small values for the maximal threshold ($\theta_{\max} \lesssim 50$)

lead to an uncontrolled permanent switching of the individuals from one task to another, leading to a very high workload (see Figure 3.5(d)).

3.3.3. Specialization in dynamic environments

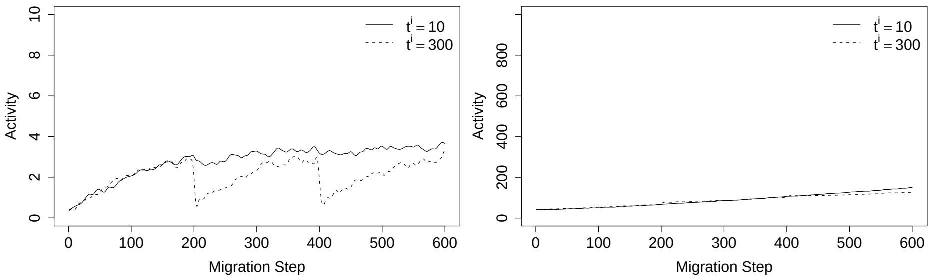
To study the influence of migration-like processes on response threshold models, three colonies of different sizes $N \in \{10, 100, 1000\}$ were investigated. The maximal threshold for each colony was chosen according to the results outlined in the last section, so that colonies could adapt properly to changing environments: $\theta_{max} = 10$ for $N = 10$, $\theta_{max} = 50$ for $N = 100$, and $\theta_{max} = 300$ for $N = 1000$. Each simulation run was repeated 10 times and in each run, 50 migrations with a migration interval length of $t^m = 200$ steps were performed. For this fixed migration interval length different demand variation interval lengths $t^\Delta = \{210, 220, 230, 300, 500\}$ were tested, which led to inter-migration intervals $t^i = t^\Delta - t^m$, $t^i \in \{10, 20, 30, 100, 300\}$. During inter-migration intervals, the demand for the migration task T_1 was set to $D_1^s = 0.3$, while during migration tasks it was set to $D_1^c = 0.9$. The demand for the other task T_2 was held constant throughout the simulation at $D_2^s = 0.8$.

An interesting aspect of the migration process is the effect of the inter-migration interval length t^i on the colony's migration task activity during the migration. Remember that within this setup, different-sized colonies use different maximal thresholds, which represent the individuals' learning behaviour. Small maximal thresholds constitute coarse learning, which means that within a few learning / forgetting steps an individual can switch from being an expert to being a complete non-expert, and vice versa. On the other hand large maximal thresholds lead to slower, more fine-grained learning.

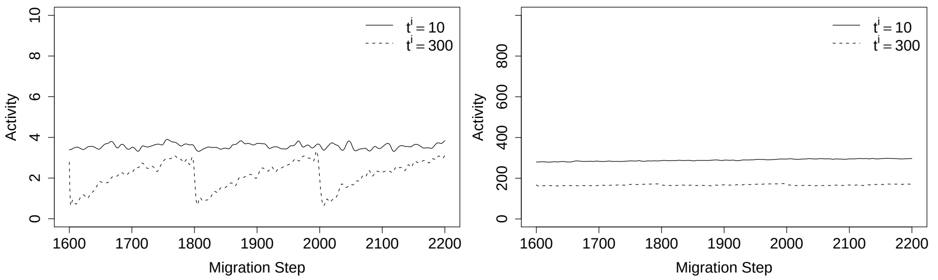
The learning behaviour has an impact on the length of the time interval an individual can sustain its experience. In terms of the migration task, this means that an individual which did sufficient work on the task to become an expert during migration will be able to sustain its threshold over a non-migration period, depending on its maximal threshold.

Figure 3.6 depicts a colony's average activity (i.e., the number of individuals that work for the task per time step averaged over all simulation runs) for the migration task in a subset of three migrations at the beginning and at a later stage of the simulation. Results for two different-sized colonies $N \in \{10, 1000\}$ and two inter-migration interval lengths $t^i \in \{10, 300\}$ are shown.

From the figure, differences in a colony's adaptive behaviour can be seen for different colony sizes (compare left and right columns of Figure 3.6). The small colony adapts quickly (due to its small maximal threshold). However it can only sustain its activity level between successive migrations if the inter-migration interval is short enough. Given a longer interval ($t^i = 300$), activity can not be sustained and the colony relearns the task



(a) $N = 10$ Migration 1 - 3 (b) $N = 1000$ Migration 1 - 3

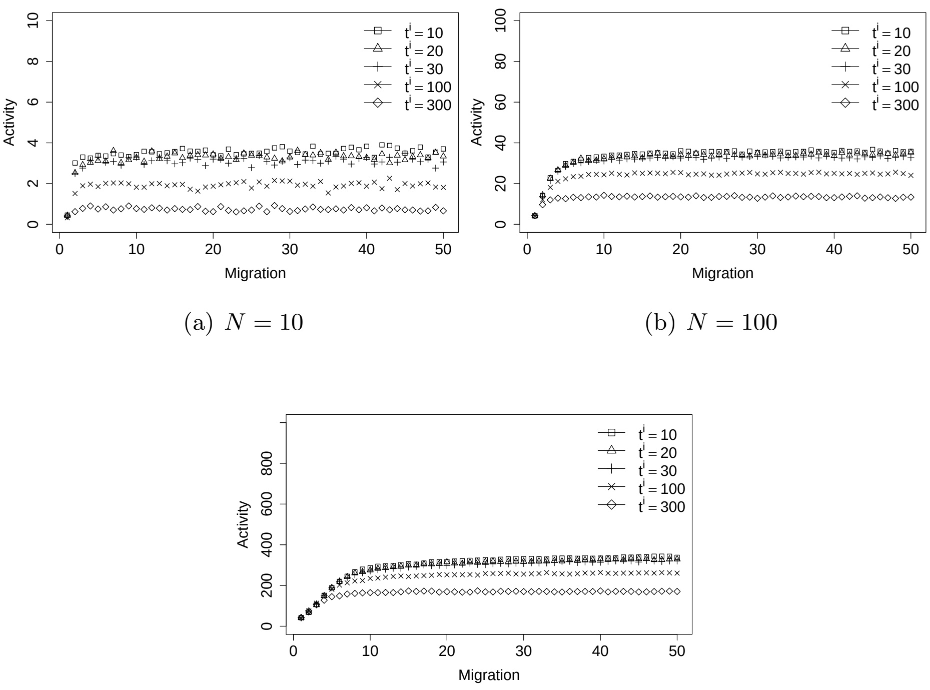


(c) $N = 10$ Migration 9 - 11 (d) $N = 1000$ Migration 9 - 11

Figure 3.6.: Colony's average activity (number of individuals working averaged over all simulation runs) on the migration task T_1 over different subsets of 3 migrations (see subfigure labels) for different-sized colonies $N \in \{10, 1000\}$ and under different inter-migration interval length $t^i \in \{10, 300\}$. Please note that the activity for the task during the inter-migration intervals is not displayed.

each migration. In contrast, the activity of the large colony increases gradually over the course of migrations, and does not show any decrease from the end of one migration to the beginning of the next, regardless of the length of the inter-migration interval. However, in the case of a long inter-migration interval the overall level of task activity is lower than was found for shorter inter-migration intervals.

The effect of sustainability is even further highlighted by looking at the activity for the migration task in the first step of each migration, which is typically the lowest activity per step during the whole migration. Since the migration task had a very low demand until the first step of the migrations, this will lead only individuals with very low thresholds for this task to work on it. When a migration starts the demand associated with the migration task T_1 is set to $D_1^c = 0.9$. As a consequence, around 45% of the population will need to work on the migration task during the migration period in order to keep the stimulus stable, which will lead to an activity increase in this task.



(a) $N = 10$

(b) $N = 100$

(c) $N = 1000$

Figure 3.7.: Activity for the migration task in the first step of each migration over the course of 50 migrations for different-sized colonies $N \in \{10, 100, 1000\}$.

In Figure 3.7, the activity for the migration task in the first step of each migration is depicted. As can be seen, the largest colony ($N = 1000$) performs best in terms of maintaining an activity onset for the migration task during long inter-migration intervals after a few migrations have been completed. In the first step of the first migration all colonies have similar performance, with an activity level at approximately 10% of the colony size.

In Table 3.3 the normalized (in terms of colony-size) mean and standard deviation of the activity for the migration task is shown for the first step of different migration intervals under different lengths of inter-migration intervals. This table illustrates that a colony's ability to maintain an onset for a task over a long period of time increases with its size. While all colonies are able to sustain their activity during the short inter-migration interval ($t^i = 10$) at a similar level, clear performance differences can be observed for the longer inter-migration interval. As outlined above, the reason for this can be found in the different maximal threshold values which are used to increase the performance of different-sized colonies in dynamic environments.

Table 3.3.: Normalized (in terms of colony-size) mean activities for the migration task at the first step of the 10th, 25th, and 50th migration for inter-migration interval length $t^i \in \{10, 300\}$ and different colony sizes N; in parentheses the standard deviation of the normalized mean activity is given.

$t^i = 10$

N/Interval	10	25	50
10	0.34 (0.15)	0.34 (0.13)	0.37 (0.15)
100	0.33 (0.04)	0.35 (0.05)	0.35 (0.05)
1000	0.29 (0.01)	0.33 (0.02)	0.34 (0.01)

$t^i = 300$

N/Interval	10	25	50
10	0.07 (0.08)	0.07 (0.09)	0.06 (0.07)
100	0.14 (0.03)	0.14 (0.03)	0.13 (0.03)
1000	0.16 (0.01)	0.17 (0.01)	0.17 (0.01)

3.4. Conclusion

In this chapter, the adaptiveness of different-sized colonies to changing environments was analysed using a standard threshold reinforcement model. The performance of different-sized colonies was compared under an identical parameter setup. While very small colonies adapt gradually to changes in the environment and thus need a comparatively long time for the adaptation process, large colonies have the opposite problem - they adapt too coarsely and fast, which creates turbulences in the system.

The reason for this was identified in the colony-size dependency of stimulus growth, which has a direct effect on colony members' perception of task requirements. The level to which a stimulus can grow in a large colony is high and an increase can thus create instant awareness for a task and over-promote it. In small colonies stimulus growth is slow and this can lead to a long period of unawareness for the task.

Since colonies should be able to adapt to environmental changes in an appropriate manner, regardless of their size, our observations suggest that individuals in different-sized colonies need different means of flexibility. The flexibility of a colony's individuals can be altered by changing the maximal threshold $\theta_{\max}$ which is used. The maximal threshold determines how large a stimulus can grow before even the individual with the worst threshold will start working on it. It also constitutes the individuals' memory and thus how many steps individuals will need to switch from being experts for a task (i.e., threshold equals zero) to being very insensitive (i.e., threshold equals maximal threshold).

The performance of different-sized colonies was tested for different maximal thresholds. Our studies show that given fixed learning and forgetting rates, small colonies are able to adapt best using small maximal thresholds. In large colonies however, small maximal thresholds cause excessive overworking, which declines with increases in the maximal threshold.

This suggests that individuals of different-sized colonies require different flexibility in their learning behaviours, here parametrized by the amount of possible learning and forgetting steps needed to successfully adapt to dynamic environments. While individuals in large colonies can afford a more gradual adaptation, individuals in small colonies need to adapt fast. This corresponds to observations in nature where small colonies are observed to contain more all-round workers, while large colonies often exhibit specialists that are only receptive to a limited range of tasks.

The second aspect investigated in this chapter is to how well different-sized colonies can sustain task-memory over consecutive task demand changes. To do so an iterative migration-like process was studied. The results show that large colonies are able to maintain an activity onset for the recurring migration task over a long period of time, while individuals in small colonies are not able to sustain low thresholds for this task over a long period of inactivity. The inability of small colonies to keep experts over a long period of absence of a task can be seen as a trade-off for their increased need for flexibility which they would lack by keeping specialists. On the other hand, large colonies are able to maintain a fraction of experts or specialists. As the standard response threshold model does not exhibit colony-size dependent specialization, our study suggests that specialization is triggered through environmental changes.

4. Influence of spatial task distribution on division of labour

Standard threshold models do not implement any form of spatial dimensionality. They assume that every individual is able to sense the global task-stimuli, and thus react to the task necessity on a colony level. While such a general setup is sufficient to study a wide range of dynamics and specialization under the response threshold hypothesis, in nature tasks as well as individuals are usually not homogeneously distributed over the sphere of action of a colony.

Spatial distribution in social insect colonies can be imposed on the colony by the environment or by the individuals themselves. Foraging is a good example for environmentally inflicted spatial task-distribution, as it requires individuals to navigate in space outside of the nest. Thus, while engaged in foraging, an individual will not be able to sense the necessity of tasks which are performed inside the nest, such as brood-care. As these tasks are spatially mutually exclusive they will impact the workers' behaviour (it is very unlikely to find a brood sorter in the foraging area).

Spatial task separation is not always imposed by the environment but can also be the result of individual behaviour, for example via stigmergy (THERAULAZ AND BONABEAU, 1999). Several social insect species exhibit task segregation in their nests, which reveals itself in the spatial organization of the nest itself. A honeybee hive, for example, contains different areas designated for brood, pollen and honey storage, thus creating a spatial distribution (JOHNSON, 2009). Another example for individually induced spatial segregation can be found in several ant species which sort their brood in a centrifugal manner around the nest centre according to the larval stage, which allows easy deployment of the different care required by the larvae (SENDOVA-FRANKS AND FRANKS, 1993). A recent study has shown that bumblebees occupy different spatial zones within their hive according to their age, and these zones are usually associated with different tasks (JANDT AND DORNHAUS, 2009). As outlined in Chapter 2, RICHARDSON ET AL. (2011) recently proposed a spatial fixed-threshold model (SFTM) and were able to show that including spatial features in response threshold models can lead to homœostasis.

These empirical and theoretical examples show that spatial distribution plays an important part in the dynamics underlying self-organization and thus division of labour in social insects. This chapter investigates the effect of spatial task-distribution on productivity in the context of response threshold models. In addition, potential "sorting strategies" that could enable a colony to maintain spatial task separation are outlined and tested regarding their adaptiveness in dynamic environments.

4.1. The spatial threshold reinforcement model

In the standard TRM which was introduced in Chapter 2 an individual is always able to encounter the stimuli of all tasks present in the system. As mentioned previously, in real insect colonies tasks as well as individuals are usually spatially distributed within the environment. Single individuals might thus only be able sense local stimuli. To include this aspect in the context of response threshold models the standard TRM was extended by spatial aspects. This extended model will be referred to as spatial threshold reinforcement model (STRM).

Formally, the STRM has l different locations $L_1, \ldots, L_l$. Each location L_k $(0 \leq k \leq l)$, contains n_k individuals. Therefore the whole system contains $N = \sum_{k=1}^{l} n_k$ individuals. For simplicity we assume that the number of individuals at each location is fixed. As outlined in Chapter 2 a standard TRM contains m tasks. Any of these m tasks can be present at each location with a certain local demand. An individual i located at location L_k can only sense the local necessity for the tasks. Thus, a demand D_j^k and stimulus S_j^k exist for each task T_j at each location L_k. Let $D_j = \sum_{k=1}^{l} D_j^k$ be the total demand for task T_j and $S_j = \sum_{k=1}^{l} S_j^k$ be the total stimulus for task T_j over all l locations. Individuals only consider the local stimuli when deciding which task to work on and the work of an individual contributes only to the tasks at its location. Hence the demand and stimulus values at the different locations are independent.

The stimulus of a task reflects the necessity for a colony to work on that task. As it is correlated with the workload fulfilment of a colony, it can also be seen as the amount of work for a given task which has accumulated without being processed until a certain time step. For example a high stimulus for feeding means that there are many starving larvae in the colony. In colonies with high stimulus values there is more undone work around than in colonies with low stimuli. Thus the stimuli values can serve as a quality measure and are used to evaluate the performance of the systems in this study. The sum of all stimulus values for all tasks at a given time step

$$S = \sum_{j=1}^{m} S_j \tag{4.1}$$

is also called the total stimulus and it is assumed that the lower this value, the better the performance of the system.

After a system has adapted to the required demand distribution, the stimuli will still display minor fluctuations over time. In order to disregard these fluctuations we average the total stimulus over an interval of time steps to yield the value $S^{\varnothing}$. Within this study, stimulus values were recorded after $5 \cdot t_{init}$, which in previous studies proved to be sufficient time for a system to adapt to a given demand distribution (see Merkle and Middendorf 2004).

4.2. Experimental setup

Unless stated otherwise the presented results are averaged over 50 simulation runs using the following parameter values. The number of locations is $l = 2$ and the number of different tasks is $m = 2$ (this could for example model an ant nest with two brood chambers where it is necessary to feed the larvae and to take care of the eggs). The number of individuals at each location L_k is assumed to be $n_k = 100$. The total demand for task T_j, $j \in \{1, 2\}$ is $D_j = 1$. For the learning parameter ξ and the forgetting parameter ϕ we take the values $\xi = 3.5$ and $\phi = 4.0$ (these values have also been used in previous studies (Gautrais et al., 2002; Merkle and Middendorf, 2004). The length of initialization phase is set to $t_{init} = 10000$ steps, the maximum age of an individual is $a_{\max} = 1000$ and the maximal threshold values are $\theta_{i,j}^{max} = 20$.

4.3. Experiments

4.3.1. Division of labour in a spatially distributed environment

In this section the effects of spatial task distribution on division of labour under the response threshold hypothesis are studied. This is a necessary first step since it is not clear whether or not spatial task distribution has any effect on the model.

Fixed demands

To investigate whether the spatial distribution of tasks has any effect on the individuals' work performance in the system, we first consider a system with a static task distribution, meaning that the demand for every task at each location is fixed. Since individuals at each location consider only the local stimuli in order to choose a task to work on, such a setup corresponds to two threshold reinforcement models running in parallel.

Such a system can be studied under different task demand distributions at each of the two locations L_1, L_2. The task distribution among the two locations can be seen as

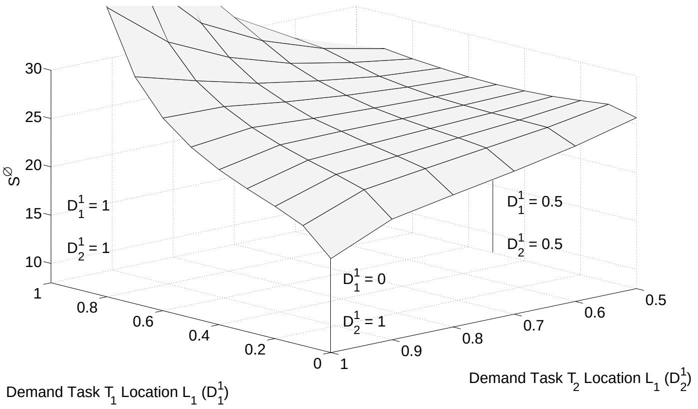


Figure 4.1.: Average total stimulus $S^{\varnothing}$ under different demand distributions D_j^k for both tasks on both locations, over 50000 time steps, with a total demand of $D_j = 1.0$ for each task.

a representation of how spatially separated the tasks are. In a setup with strong spatial separation, each task is only present at one location (e.g., $D_1^1 = 0, D_2^1 = 1, D_1^2 = 1, D_2^2 = 0$). In contrast, in a system where the task demands are not spatially separated the tasks are equally present at each location (i.e., $D_1^1 = D_1^2 = 0.5$, $D_2^1 = D_2^2 = 0.5$).

The system should be able to cope with any demand distribution when the demand-sums for both tasks are fixed to $D_j = 1.0$. Even when the tasks are only present at a single location this can be accomplished by workers at the location, even though they would need to work with full force, since they would need to provide maximal work for each task at each timestep to maintain a constant stimulus value at their location.

Figure 4.1 depicts the average total stimulus $S^{\varnothing}$ of the system under various distributions of demands on the two locations. As can be seen, the system is able to keep the total stimulus on the lowest level in a scenario where both tasks are only present at different locations. In this case the individuals at each location can focus on a single task only, since demand and thus the stimulus for the other task are zero.

Another observation that can be made is that the total stimulus increases with decreased separation (i.e., the demands of both tasks become more similar at each location) and with a more unequal distribution of the tasks at the two locations (i.e., the sum of the demands at both locations becomes unequal). The stimulus reaches its maximum in the situation where both tasks are present only at the same location. As each individual in the system has a fixed spatial location, in this case half of the colony has all the workload while the other half of the colony is idle.

A colony can cope with any possible spatial distribution of demand in situations where $D_j \leq 1.0$ holds for each task. This is due to the fact that the number of work δ_j a task requires per timestep is defined as $\delta_j := D_j \cdot W_{max}$, where D_j constitutes the demand D_j of task T_j and W_{max} denotes the maximal amount of work that can be performed on average on a task (see Chapter 2 for more information). Accordingly the demand value for each task was defined as $0 \leq D_j \leq 1$ in Chapter 2. As the total demands D_j used here are distributed over two locations, we can however assume higher total demand values. Even in a situation where $D_j > 1.0$, there are still possible distributions of the demands on the two locations which fulfil $D_j^k \leq 1$. This is not the case with increasing D_j. For example with a total demand of $D_j = 2.0$, for a given task T_j, only one distribution is possible in a system with 2 locations: full demands on both sites for that task.

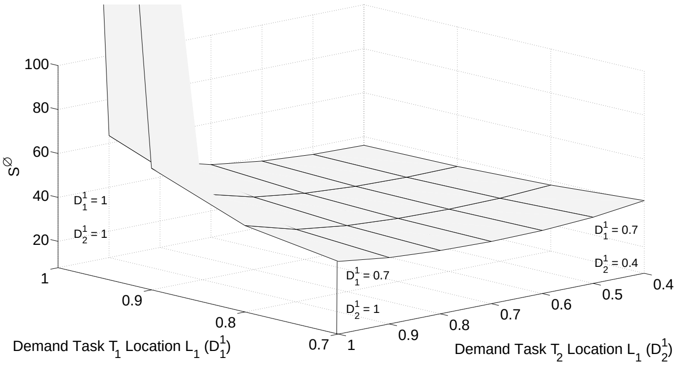


Figure 4.2.: Average total stimulus $S^{\varnothing}$ under different demand distributions D_j^k for both tasks on both locations, over 50000 time steps, with a total demand of $D_j = 1.4$ for each task.

Figure 4.2 depicts the average total stimulus value of a system with the total demand of each task fixed to $D_j = 1.4$. As in the previous experiment the system is able to achieve the best average stimulus value in a situation where the demands of the two tasks are spatially separated over two locations. Note that a complete separation is not possible any more, since this would require task-demands greater than 1.0 at each location. However the experiment shows that even partial sorting can increase the performance of a system. It thus seems that a spatial task distribution is beneficial in a multi-location system if the total amount of demand permits it (i.e. $D_j \leq 2.0$ in the case of 2 locations).

Impact of demand redistribution

A second aspect that is interesting regarding spatial task distribution is whether a redistribution during runtime will affect the system in any way. We thus consider a scenario where the demands of the tasks can change at both locations, to investigate if and how fast the system can adapt the total stimulus after a demand change. One would expect that the stimulus values change after a change in the demand values, since they repre-

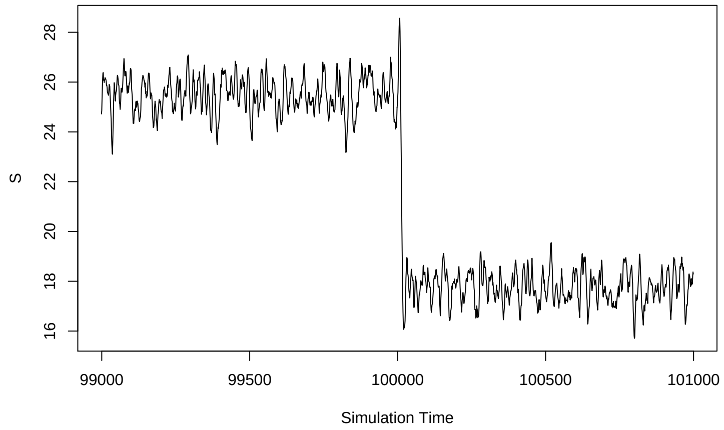


Figure 4.3.: Change of stimulus in a system shown for 2000 simulation steps after step 99000; at simulation step 100000 a spatial separation between the tasks is enforced by changing demands from $D_1^1 = D_1^2 = D_2^1 = D_2^2 = 0.5$ to $D_1^1 = 0, D_2^1 = 1, D_1^2 = 1, D_2^2 = 0$.

sent the necessity of a task. Given a high demand that is then decreased, the associated stimulus should also decrease until it attracts the needed amount of individuals, because the high stimulus will attract too many workers for the task and thus decreases due to over-work. If the stimulus can decrease after a demand separation, this would indicate that a proactive rearrangement of task demands can effectively increase the performance of such systems. This might also explain why such phenomena are often observed in nature, for example in ant colonies where the ants arrange larvae and eggs at different locations (Sendova-Franks and Franks, 1993).

The simulation started with suboptimal task distribution with respect to the total stimulus. At timestep 100000 the task demands were changed to be spatially more separated. The result are depicted in Figure 4.3. The total stimulus varies between 23 and 28 during the first phase with a demand setting of $D_1^1 = D_1^2 = D_2^1 = D_2^2 = 0.5$. After time step 100000 when the demands were changed to $D_1^1 = 0, D_2^1 = 1, D_1^2 = 1, D_2^2 = 0$, the total stimulus decreases very fast and varies afterwards between 15 and 20. This indicates that demand rearrangements during a simulation can indeed have a beneficial effect and increase the performance of a system.

4.3.2. Individual-based task redistribution

As we have seen, spatially separated tasks within a spatial threshold reinforcement model can increase the overall performance of a system. In addition, a redistribution of tasks

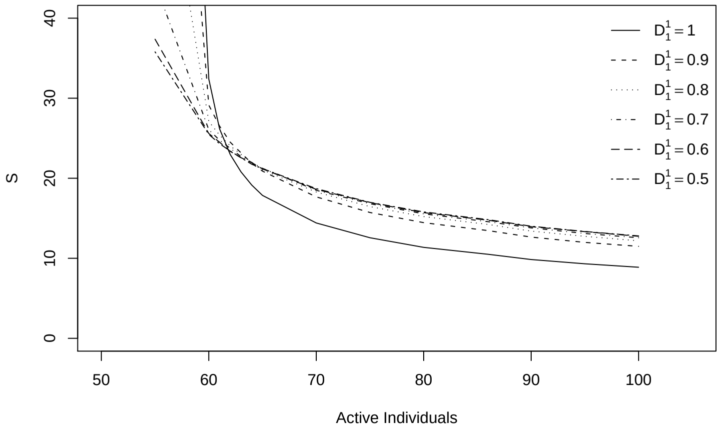


Figure 4.4.: Total stimulus for different number of active individuals under different demand distributions D_j^k. Total number of individuals at a single location was $n_k = 100$.

to increase their spatial separation increases a system's efficiency. Here we investigated whether it is efficient for a colony to change the spatial distribution of tasks by means of active redistribution carried out by part of its workforce. Clearly, an active rearrangement of the tasks by the colony itself comes with additional costs as it requires individuals to spend part of their time working on task rearrangement. In the following we will refer to tasks rearrangement as task sorting. Individuals that redistribute tasks are accordingly called sorting individuals. Tasks are referred to as sorted if each task occurs only at one location and no other task occurs there. As sorting comes with a cost we will first investigate how much workforce a colony can devote to task-sorting under aspects of work efficiency.

Figure 4.4 depicts the influence of the number of active individuals on the total average stimulus for task T_1 under different demand distributions. The other individuals are assumed to be passive, which means that they will not work. It can be seen that in an environment where tasks are unequally distributed over locations (e.g. $D_1^1 = 1$), the colony can keep the total stimulus at a lower level when some of the individuals remain passive (i.e., do not work), compared to a situation where the tasks are equally distributed over the locations and all individuals are active (i.e., potentially work for a task).

This changes however when the proportion of active individuals drops to less than about 60%. In this case, colonies in less unequally distributed task environment are able to maintain better average stimulus levels. This is due to the fact that in the threshold reinforcement model an individual encounters each task with the same probability, which means that each task is chosen with a maximum probability of 0.5 (see Eq. 2.2 in Chapter 2 for more details).

Consider a situation where both tasks are present at one location only, again an individual will only be able to choose a task with the probability of 0.5. However, if it does not choose the first task it will definitely choose the second, so it will work with a probability of 1.0. If in contrast the tasks are sorted, only one task will be present at a given location and the stimulus for this task at the respective location will be high. However, agents will only start to work for this task with a probability of at most 0.5.

Besides these extreme cases, the results depicted in Figure 4.4 indicate that under the response-threshold hypothesis it is beneficial for a social insect colony to spare some workers from the actual working process and use them to maintain a favourable spatial distribution of the tasks. In the the rest of this section we will discuss how active sorting can be integrated into the spatial threshold reinforcement model.

Task rearrangement by a specialized caste

In many social insects species, colonies are composed of multiple behavioural and or morphological castes (Wilson et al., 2009), i.e., groups of individuals that manage different duties in the colony. Task rearrangement could be accomplished by a specialized cast of individuals that focuses on this rather than on the "normal" tasks. Hence, let us consider that the current model contains two castes: workers and task-sorters.

Formally a fraction f_{sort} of the individuals at every location constitutes the members of the sorting caste and the rest of the individuals are workers. Sorters solely focus on sorting (details are given later) and do not contribute to the normal workforce. Unless stated otherwise, a value of $f_{sort} = 0.1$ was used for the simulations.

In a realistic scenario the amount of demand that a task-sorter can move from one location to another will be limited, e.g., an ant can carry only one egg from one brood chamber to another at a given time. Therefore, it is assumed here that within one time step a task-sorter can transport from one location to another exactly the amount of task demand $W_{pickup} = m/n$ that can be accomplished by a worker in a single simulation step. Different sorting strategies are possible for the task-sorters to decide which task to transport from which source location to which sink location. Two sorting strategies are proposed and investigated here:

i) *Deterministic sorting.* For this strategy it is assumed that the task-sorter knows which task belongs to which location and try to shift misplaced demands accordingly. More formally: each task T_j is assigned to a location L_k. In every time step each task-sorter at a location L_k will try pick up a demand for a task that is not assigned to location

L_k and place it on the assigned location of the tasks. This is done by the agents without changing their location.

ii) *Stigmergic sorting.* Stigmergy mechanisms use individual-based changes to the environment as feedback that triggers the behaviour of other individuals. Many aspects of social insect behaviour rely on stigmergic principles (THERAULAZ AND BONABEAU, 1999). Since stigmergy could also be useful in the context of spatial task rearrangement, a strategy is adopted which is similar to that used by ants to aggregate brood (SENDOVA-FRANKS AND FRANKS, 1993). This aggregation strategy is based on the density of brood items, e.g., eggs or larvae. The probability for an unladen agent to pick up an item is higher the lower the number of other items in the neighbourhood. On the other hand the probability to drop an item again is higher the more objects are around. Given a random distribution of items in an environment, such a strategy leads to the formation of small clusters of items which will slowly merge into bigger item clusters.

In the stigmergic sorting strategy a task-sorter becomes active at each time step. The probability that a task-sorter at location L_k will pick up demand from task T_j is

$$P^k_{j,pickup\_from} = \frac{P^k_{j,pickup}}{\sum_{j=0}^{m} P^k_{j,pickup}}, \qquad with \quad P^k_{j,pickup} = \left(\frac{T_{pickup}}{T_{pickup} + S^k_j} \right)^2 \tag{4.2}$$

where S^k_j denotes the stimulus of task T_j at location L_k and T_{pickup} is a parameter that is called pickup threshold. As can be seen, the smaller S^k_j is, the more likely it is that the task-sorter picks up demand of task T_j.

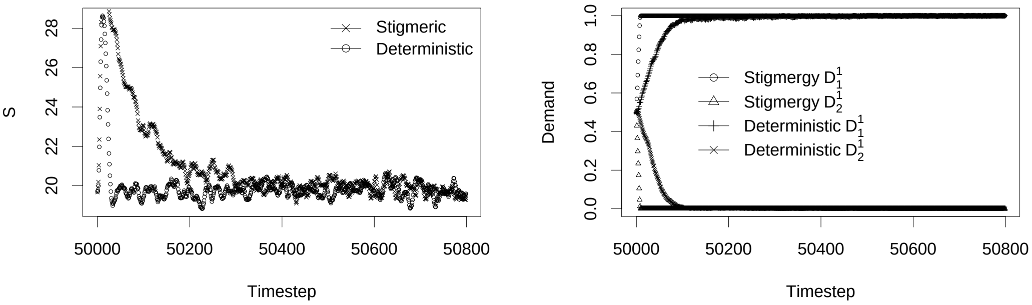
Once a task demand is picked up, the task-sorter decides at which location to drop it. Again the stimulus values of the corresponding task at different locations determine the probability to drop the demand at location L_k

$$P^k_{j,drop\_at} = \frac{P^k_{j,drop}}{\sum_{k=0}^{l} P^k_{j,drop}}, \qquad with \quad P^k_{j,drop} = \left(\frac{S^k_j}{S^k_j + T_{drop}} \right)^2 \tag{4.3}$$

where S^k_j denotes the stimulus of task T_j at location L_k and T_{drop} is a parameter that is called the drop threshold.

Figure 4.5 depicts the efficiency of a system with task-sorters over 800 consecutive simulation steps after a change in demand. This system used the standard setup outlined in section 4.2 and a sorter fraction of $f_{sort} = 0.1$ (i.e., 10 task-sorters per location).

As can be seen, both sorting methods achieve a spatial separation of tasks which results in a lower total stimulus. In the first step the task stimuli increase for both sorting-methods (Figure 4.5(a)). This is due to the fact that at step 50000 demands D^k_j are reset to 0.5. This creates turbulences since workers have to adapt to the new demand situation while



(a) Total Stimulus

(b) Demand

Figure 4.5.: Change of total stimulus and demands at the different locations in a system with task-sorters shown over 800 simulation steps after step 50000. Demands D_j^k are set to 0.5 at time step 50000 and $f_{sort} = 0.1$ resulting in 10 sorters at each location given the standard setup of $n = 100$ individuals at each location.

sorters have to start to redistribute the demands. The deterministic sorting method is able to reach the spatial task distribution faster then the stigmergic sorting method. This is not surprising, since in the deterministic case, sorting individuals intrinsically know on which location to place which task demand. This however can also be a disadvantage in dynamical situations where rearrangement needs to be flexible.

To investigate sorting performance in a dynamical environment it is assumed that at each time step t a certain fraction f_{cease} of each task demand D_j^k ceases at each location. In order to keep the total demand D_j for each task T_j constant, this demand has to be fed back into the system. $f_{re-enter}^{k,j}$ denotes the fraction of the vanished demand $f_{cease} \cdot D_j$ placed on location L_k, and the fraction of the demand placed on the other location thus corresponds to $1 - f_{re-enter}^{k,j}$.

Within such a system the task demands on each location will slowly converge towards the distribution given by $f_{re-enter}^{k,j}$. How fast this happens depends on f_{cease}. If for example $f_{cease} = 1.0$ it would only take one time step. Depending on $f_{re-enter}^{k,j}$, sorters might have to work at each time step in order to maintain a spatial task distribution.

Figure 4.6 depicts the total stimulus for three systems under different cease fractions f_{cease} and rearrangement fractions $f_{re-enter}^{k,j}$, one without task rearrangement, one using deterministic sorting and a one using stigmergic sorting. Additionally the pairwise differences between the total stimulus of these systems with differing sorting methods are shown. Please note that the rearrangement fractions for the different tasks sum to 1. The data presented were averaged over the last 500 steps of 10 simulation runs lasting for 100000 time steps.

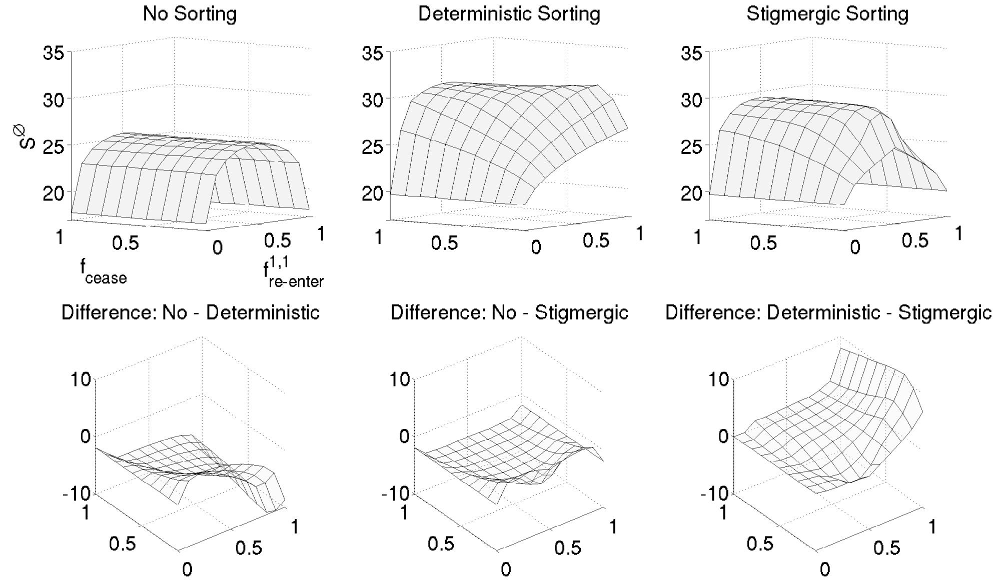
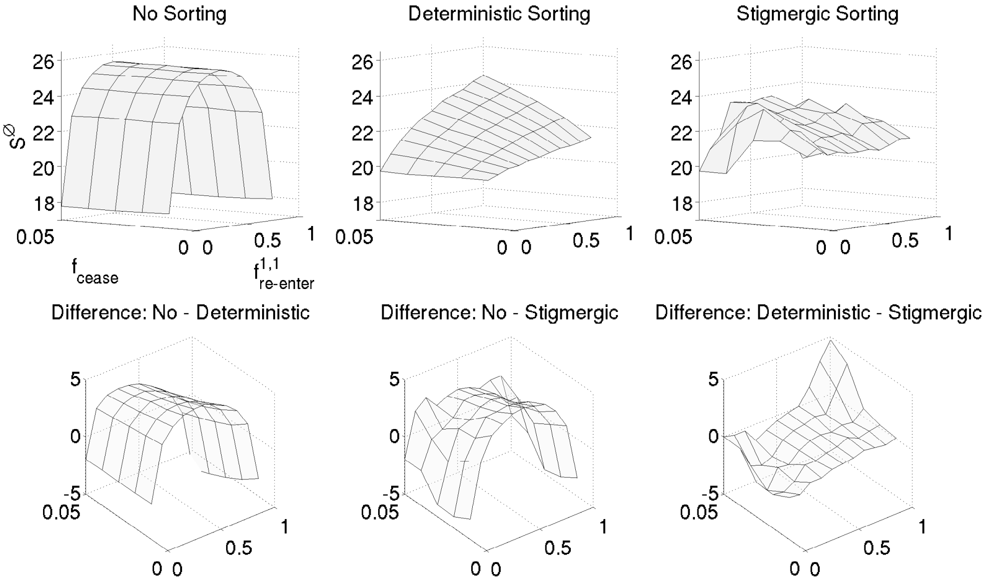


Figure 4.6.: Total average stimulus values and total average stimulus differences for different sorting methods as a function of different cease f_{cease} and rearrangement fractions $f^{k,j}_{re-enter}$.

The figure clearly depicts the lack of flexibility of the deterministic sorting method. Its inability to adapt the sorting to locations which would be beneficial makes its performance worst in scenarios where the tasks appear mainly on locations where they are not expected to be. Stigmergic sorting can recognize such situations and adapt the sorting accordingly.

As can be seen, both sorting methods are only able to achieve lower stimuli levels than a system with no sorting agents if the cease fraction f_{cease} per time step is low. This indicates that sorting in a spatial system is only beneficial if the amount of task demand that needs to be moved at each time step can be handled by the colony. Otherwise sorting has a negative effect and results in higher stimuli levels.

Figure 4.7 depicts the total stimulus for systems with small cease fraction f_{cease}. In such scenarios systems using sorting will always outperform systems without task sorting, except in cases where the external demand change already results in a well sorted system. The reason is that in such cases a system without task sorting mechanisms has more workers available.


No Sorting

Deterministic Sorting

Stigmergic Sorting

Difference: No - Deterministic

Difference: No - Stigmergic

Difference: Deterministic - Stigmergic

Figure 4.7.: Total average stimulus values and total average stimulus differences for different sorting methods as a function of different cease f_{cease} and rearrangement fractions $f^{k,j}_{re-enter}$.

Task rearrangement as a third task

The sorting methods considered before have the potential drawback that the number of task-sorters is fixed and can not adjust itself according to sorting requirements. Thus, it might be more advantageous for a system to use one unifying principle that incorporates "task sorting" as a task in itself. However, it is not obvious how to assign a reasonable stimulus value to the sorting task, as it lacks a demand value. As mentioned earlier, sorting should ideally result in a spatial task separation. As such, given the sorting task T_{j+1}, we suggest that the stimulus for this task on location L_k is given by

$$S^k_{j+1} = S_{j+1,max} \cdot (\min_{j=1}^{m} S^k_j / \max_{j=1}^{m} S^k_j) \tag{4.4}$$

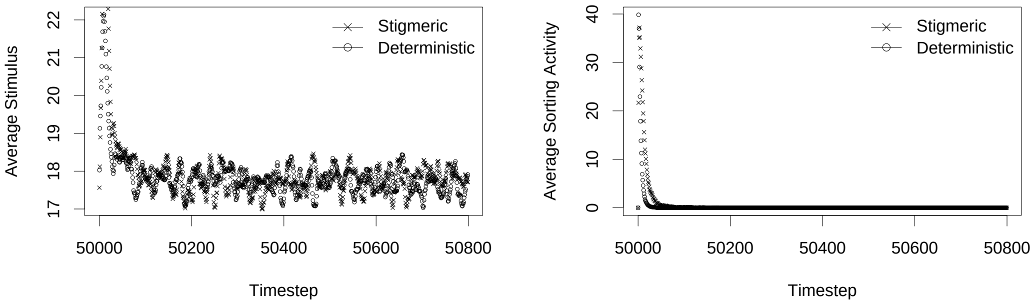
where $S_{j+1,max}$ is a parameter that denotes the maximal sorting stimulus. As a task's stimulus values S^k_j can be sensed under the threshold reinforcement model, it might also be plausible for many real systems that the relation between stimuli can be estimated by the individuals, at least when the number of tasks is not too large.

Clearly, for systems with $m = 2$ tasks the proposed stimulus measure will result in high values if both normal (i.e. non-sorting) tasks have approximately equal stimuli at a single location, and will result in low values if only one task has a high stimulus. Regarding the underlying sorting mechanism, we re-apply the methods proposed before. In contrast to the morphological sorting approach, the standard threshold equation is now used to determine whether an individual will work as a sorter

$$P(X_i = j) = \frac{1}{m} \cdot \frac{{S_j}^2}{{S_j}^2 + {\theta_{ij}}^2} \tag{4.5}$$

An individual's sorting duration and idle times are also determined as for the other tasks. Only the decision of which task is to be sorted is based on the sorting mechanisms outlined earlier (i.e., deterministic / stigmergic sorting).

Care must be taken when models with and without sorting tasks are compared with respect to their efficiency. This is due to the fact that the introduction of a sorting task affects the stimulus values. All tasks are encountered with the same likelihood, and the additional sorting task affects the maximal amount of work W_{max} that can be done for a task on average, and thus stimulus growth. Hence, the total stimulus values can not be used as a direct comparison of efficiency between models with different amounts of tasks. Figure 4.8 depicts the efficiency of a system that incorporates sorting as a third task over 800 consecutive simulation steps after a change in demand. As can be seen, the system can achieve a spatial distribution of tasks, which also leads to a decrease in the total stimulus. Furthermore as depicted in Figure 4.8(b), work that is done for the sorting task decreases after a spatial distribution is reached (due to the sorting stimuli being 0 in such a situation). Such a system thus appears to be more flexible than the morphological sorting presented above, since a colony can adapt not only to regular task demands but



(a) Total stimulus (b) Sorting Activity

Figure 4.8.: Change of total stimulus and demands on the different locations in a system with a sorting task, shown for 800 simulation steps after step 50000; demands D_j^k are set to 0.5 at time step 50000.

also to the need for task rearrangement. This should also be evident from the average stimulus values in dynamic situations (described in the last section).

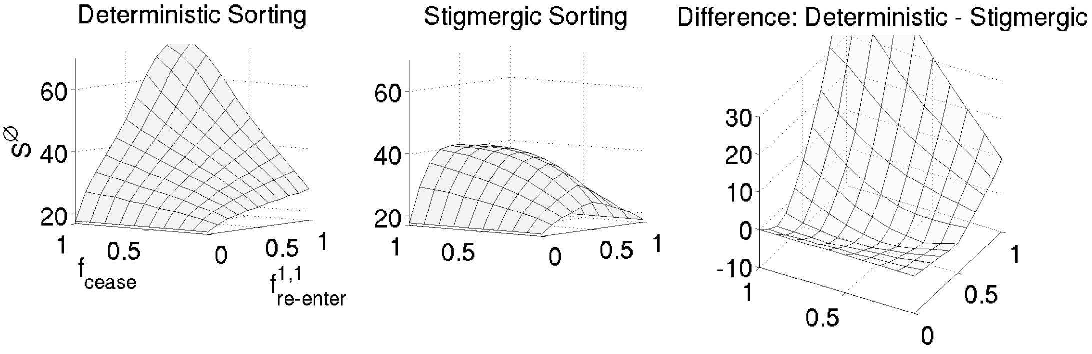


Figure 4.9.: Total average stimulus values (left, center) and total average stimulus differences (right) for different sorting methods under different cease f_{cease} and rearrangement fractions $f^{k,j}_{re-enter}$.

In a final test, the system's behaviour in a dynamic environment was tested. Figure 4.9 depicts the behaviour of the system with a sorting task as a function of different cease fractions f_{cease} and rearrangement fractions $f^{k,j}_{re-enter}$. Even though the stimuli values can not be directly compared, comparing the topology of the total stimulus values in Figure 4.9 and Figure 4.6 suggests that a system with sorting as a third task is able to cope with stronger changes in external task demand (i.e., f_{cease}, $f^{k,j}_{re-enter}$) than a system with a fixed amount of sorting individuals. This is due to the fact that a system that implements "task sorting" as a third task is not restricted to a fraction of sorters, but can dynamically adjust its workforce to a given situation. However given that each task is encountered with the same likelihood, even such a system has its sorting boundaries. These are encountered when the amount of demand that needs to be rearranged per time step is greater then the maximal amount of demand that can be rearranged per time step.

4.4. Conclusion

Threshold reinforcement models constitute a method for explaining and analysing division of labour in social insect colonies. While these models are able to capture many aspects of division of labour in social insects, they do not incorporate any form of space, which means that each individual is able to sense the entire colony's need for any task at any time. In nature, however, space and the spatial separation of tasks and individuals is ubiquitous. While such a spatial separation is often imposed on a colony by its environment (e.g., foraging areas are not located in the hive), it has been shown that individuals themselves establish and maintain spatial separations.

To study the effect of different spatial distributions on division of labour, we introduced an extension of the threshold reinforcement model, the so-called “spatial threshold reinforcement model”, which incorporates spatial task distribution across multiple locations into these models. In the context of this model it was shown that colonies in spatial environments achieve their best performance in situations where tasks are completely spatially separated (i.e., each task is exclusively present at one location). This corresponds to the aforementioned empirical observations.

As social insects are always located in a dynamic environment, they will need to actively maintain spatial separation. We investigated different ways in which a colony can actively achieve task sorting. These methods either operate in parallel to the threshold reinforcement model, which could correspond to a caste system, or are incorporated into the threshold response model as an additional task. As we have shown, both approaches can successfully sort the tasks and can approximately preserve the sorted state in a dynamic environment with external changes in task demands. However, the proposed caste system leads to a lack of flexibility as the colony can not adjust the amount of work that is devoted to task sorting. In contrast, task-sorting methods that are based on the response threshold hypothesis, in the sense that sorting itself is considered as a task, can adjust the amount of work for sorting and are thus more flexible and efficient.

In conclusion, the sorting of tasks seen in social insects can be understood in terms of the proposed spatial threshold reinforcement model as a spatial separation of tasks, which leads to better performance in such model systems.

Part II.

Honeybees as a model of self-organization in biological systems

A queen honeybee solicits food from one of her daughter worker bees (*Apis mellifera*). Photo by Alexander Wild, reproduced with kind permission[1].

> **Brad Crane:** *Are you endowing these bees with human motives? Like saving their fellow bees from captivity, or seeking revenge on Mankind?*
>
> **General Thalius Slater**: *I always credit my enemy, no matter what he may be, with equal intelligence.*
>
> *The Swarm*, 1978

Honeybees are eusocial insects that are part of the *Apidae* family. Eusociality constitutes the highest form of social organization in the animal kingdom, where reproduction is preformed by a single individual (i.e., the queen), while the necessary work is performed by a sterile caste of female workers (Wilson and Hölldobler, 2005).

A typical honeybee colony is composed of a single queen and up to several thousand workers. Although a colony lacks a central control mechanism, it is nevertheless able to solve complex tasks by means of collective behaviour. Such tasks include the division of labour among the workers (Robinson, 1992), the maintenance of a constant temperature in the hive (Jones et al., 2004), keeping track of changing foraging conditions (Beekman and Ratnieks, 2000; Beekman et al., 2004; Visscher and Seeley, 1982) and selecting the best possible nest-site (Seeley and Buhrman, 2001).

Due to their decentralized collective behaviour, honeybees have become an important model system in the field of swarm intelligence which was outlined in chapter 1. Several al-

---

[1] www.alexanderwild.com

gorithms based on honeybees' collective behaviour have been developed and applied to various domains such as network routing (Wedde and Farooq, 2006), robotics (Srinivasan, 2011) and multi-agent systems and optimization (Karaboga and Akay, 2009). Chapter 8 provides an overview of current bee-inspired optimization algorithms.

The aim of this chapter is twofold. First we will use simulation and modelling techniques to study two closely related behaviours found in honeybees: nest-site selection and a swarm's migration behaviour thereafter. Then we will investigate to what extent the bees' nest-site selection behaviour is applicable to optimization.

Chapter 5 provides an overview of the biological principles underlying the self-organizing behaviours found in honeybees. Chapter 6 introduces a spatial nest-site selection model, which will be used to study the impact of spatial nest-site topologies on the nest-site selection process. In Chapter 7 the flight guidance of honeybees will be studied. Using a guidance model, two different forms of guidance (active and passive) will be investigated theoretically. Their characteristics will be compared to those found in real honeybee swarms. In addition, this Chapter will investigate to what extent directional dissent can be incorporated in a moving group under active guidance. Chapter 8 discusses the potential of the nest-site selection process in the design of optimization algorithms and introduces the Bee Nest-Site Selection Scheme (BNSSS) which can be used for optimization. Based on this scheme the first nest-site inspired algorithm Bee-Nest will be introduced and its performance will be tested in the domain of molecular docking.

5. Biological background of self-organization in honeybees

This chapter briefly outlines the biology of the European honeybee species *Apis mellifera*, focusing on the self-organization that can be observed in honeybees and is of interest for swarm intelligence and bio-inspired computation. In particular we will describe the processes by which honeybees divide labour, forage for resources and choose a new nest-site. The interested reader should refer to Seeley's marvellous book on honeybees "Honeybee democracy" (SEELEY, 2010) or Winston's "The Biology of the Honey Bee" (WINSTON, 1987) for more detailed information on the biology underlying these fascinating creatures.

5.1. Division of labour

The queen is normally the only individual that reproduces in the colony, whereas the workers need to perform a number of tasks such as cleaning the nest, foraging for food and feeding the brood. As a colony is constantly exposed to a dynamic environment, division of labour among the workers needs to be flexible and adaptive to quickly adjust to the current needs of a colony and thus guarantee its survival.

The response threshold concept, which has been discussed in the first part of this thesis (see Chapters 2,3,4) is thought to be one of the key principles underlying the honeybees' division of labour. Thus individuals will work for a task if they perceive it to be necessary to be dealt with. Several factors have been identified which influence the receptiveness of a worker-bee towards a certain task, such as its age (SEELEY AND KOLMES, 1991), physiological state (AMDAM AND OMHOLT, 2003), social interaction (BESHERS ET AL., 2001) and genotype (OLDROYD AND FEWELL, 2007).

In many social insects an age-related division of labour can be observed, where young individuals perform tasks within the nest and older individuals take over outdoor tasks such as foraging (CALDERONE, 1998). This is also the case for honeybees, where young individuals are predominantly engaged in brood care and other nest duties, while older workers leave the hive to forage for resources.

From an evolutionary perspective, such an age-based division of labour is beneficial for a colony. This is due to the fact that outdoor tasks are usually associated with higher mortality rates, thus with only older individuals of a colony engaging in such tasks the average life expectancy of an individual will increase, increasing its average contribution to the colony (Tofilski, 2002).

The reasons for this seemingly age-dependent division of labour in honeybees is still debated. Seeley and Kolmes (1991) observed a cohort of bees within an observation hive. They reported two possible age-related task transitions among this group regarding hive duties (however only agreed on the validity of the first transition) and thus concluded that "..age polyethism for hive duties in honey bees is a reality, not an illusion" (Seeley and Kolmes, 1991, p.296). In contrast, Robinson et al. (1989) have shown that division of labour can also be observed in single-cohort colonies (i.e., a colony comprised of workers with the same age) and seems to be mediated by a juvenile hormone. Later studies (Huang and Robinson, 1992) have linked the levels of juvenile hormone within a hive to social interaction between workers, which thus also seems to impact division of labour.

In addition the physiological state of an individual has also been found to play a crucial role regarding its work (Janmaat and Winston, 2000; Woyciechowski and Moroń, 2009). Janmaat and Winston (2000) have shown that that bees that were deprived of pollen during their larval period show a different foraging onset than those who were not. In a recent study, Woyciechowski and Moroń (2009) linked the onset of foraging with the life expectancy of individuals, as they showed that individuals with a lower life expectancy (experimentally induced via CO_2 or fungus) start to forage earlier than individuals of the same age with a higher life expectancy.

During their maiden flight, queens mate with an average of 20 or so males (Palmer and Oldroyd, 2000) and store a lifetime's worth of sperm in a sperm storage device called a spermatheca. This polyandry causes a genetic differentiation among the workers within a colony – when workers that do not share the same father differ in their task threshold, genetically based task specialization results. A range of task thresholds have been found to be affected by an individual's genotype such as undertaking, guarding and foraging choice (Robinson, 1992). Imagine that workers sired by father A will be the first to start removing dead nest-mates from the hive whenever they encounter them. Because their action reduces the undertaking-stimulus for the whole colony, other individuals' undertaking threshold (e.g., those not sired by father A) will not be reached, thus leading to a specialization of a specific patriline towards undertaking. If however individuals of this patriline are removed from the nest, the undertaking stimulus will increase resulting in individuals with higher undertaking thresholds to perform the task. Such genetic diversity and the associated task-threshold differentiation among workers is thought to enable a colony to respond resiliently to changes in the environment (Oldroyd and Fewell, 2007). Empirical work has shown that honeybee colonies comprising a genetically diverse work force indeed perform better (Jones et al., 2004; Mattila et al., 2008; Mattila and Seeley, 2007).

As outlined earlier division of labour in honeybees (or rather in social insects in general) depends on many different factors rather than on a single regulatory mechanism. This adds robustness to division of labour and enables a colony to respond appropriately to a wide range of situations which require different actions.

5.2. Foraging behaviour

There are several external resources a honeybee colony requires, such as pollen, nectar and water (Winston, 1987). While pollen is used rather rapidly as it is fed to the developing brood, nectar is stored (becoming honey) to allow the colony to survive periods when forage is not available. While a colony is usually able to acquire its required water via nectar, in times of increased heat or low nectar collection individuals additionally need to forage for water to cool the hive or guarantee sufficient water levels for the colony (Kühnholz and Seeley, 1997). The foraging choice of an individual depends on its patriline as well as on experience (Scheiner et al., 2004). Sucrose responsiveness acts as a threshold, whereby individuals that are very sensitive to sucrose will forage for water if necessary, while individuals that are very insensitive to sucrose will only forage for pollen. As pointed out above the polyandry within a bee colony leads to an inhomogeneous distribution of initial sucrose sensitivity and thus a differentiation among foragers.

Gathering resources is crucial for a colony, as only colonies that contain sufficient honey stores are able to survive the winter (Seeley and Visscher, 1985). This has led to the the evolution of a unique recruitment mechanism that allows foragers to recruit nest-mates to discovered food sources: the bees' dance language (von Frisch, 1967). The use of the dance language enables a colony to rapidly exploit and monopolise profitable food sources while effectively ignoring those that are of mediocre quality (Beekman and Lew, 2008).

The honeybees' dance encodes information about the direction and distance of the food source found. Up to 7 dance followers (Tautz and Rohrseitz, 1998), known as potential recruits, are able to extract this information upon which they will leave the colony and try to locate the advertised food source. During a typical dance the dancer strides forward vigorously shaking her body from side to side (Tautz et al., 1996). This is known as the "waggle phase" of the dance. After the waggle phase the bee makes an abrupt turn to the left or right, circling back to start the waggle phase again. This is known as the "return phase". At the end of the second waggle, the bee turns in the opposite direction so that with every second circuit of the dance she will have traced the famous figure-of-eight pattern of the waggle dance (von Frisch, 1967).

The most information-rich phase of the dance is the waggle phase. During the waggle phase the bee aligns her body so that the angle of deflection from vertical is similar to the angle of the goal from the sun's current azimuth. Distance information is encoded in the duration of the waggle phase. Dances for nearby targets have short waggle phases, whereas dances for distant targets have protracted waggle phases.

Honeybees modulate their waggle dance depending on the profitability of the food source found – the more profitable the food source, the "livelier" and longer the dance (Seeley et al., 2000). As a result, bees dancing for highly profitable sites attract more dance followers than those that dance for mediocre sites. The dance language enables a honeybee colony to track the constantly changing foraging conditions (Beekman and Ratnieks, 2000; Beekman et al., 2004; Visscher and Seeley, 1982).

While the dance communication has been shown to be crucial for a colony in situations where resources are few in number and of poor and variable quality (Dornhaus et al., 2006; Grüter and Ratnieks, 2011), its importance in other situations is still debated (Grüter and Farina, 2009). One main criticism regarding the status of the waggle dance as the main mechanism underlying foraging is that it is very imprecise (Riley et al., 2005). Foragers have been found to use private information (i.e., memories of their previous foraging experience) to find and exploit resources (Beekman, 2005). Not only are foragers able to remember previous resource locations, they have also been found to memorize blooming times of plant species and exploit them accordingly (Ribbands, 1949)[1]. In addition, the waggle dance has been found to modulate other behaviours which are not directly linked to the dance information but stimulate foraging. While dancing foragers release chemical cues during their dance which stimulate the lift-off of idle individuals (Thom et al., 2007), followers have been found to take up scent cues from dancers which help them to find the advertised food patches or remind them of previously visited patches (Srinivasan and Reinhard, 2009).

As for division of labour, many different mechanisms seem to influence and steer the foraging behaviour of honeybees, enabling a wide range of appropriate foraging strategies in a dynamical environment, which is crucial for the survival of a colony.

5.3. Nest-site selection

When a bee colony reaches a certain size, workers will start feeding several female larvae a special diet of royal jelly (Winston, 1987). As a result these larvae develop into young queens. Once the new queens are ready to emerge from their cells, the old queen and

---

[1] On a side-note, bumblebees have recently been found to readjust their foraging route when incorporating new feeding locations, and can thus solve the well-known NP-complete travelling salesperson problem (Lihoreau et al., 2010)

about two thirds of the workers will leave the old colony and cluster temporarily on a tree branch or a similar structure while one of her daughter queens inherits the old nest. The cluster of bees containing the old queen is referred to as a reproductive swarm (Winston, 1987). The swarm is now homeless and needs to locate and evaluate potential nest sites – such as hollows in trees or crevices in buildings – and choose the best among several options.

A reproductive swarm of honeybees deciding on a new home is one of the most impressive examples of decentralised decision-making in animals, as only about 5% of the bees in the swarm take part in the decision-making process (Seeley et al., 1979). Several hundred scout bees fly from the swarm cluster to search for tree cavities and other potential dwelling places. The dozen or so scouts that find suitable cavities assess the quality of the site for characteristics such as volume, height, aspect of the entrance, and entrance size (Seeley and Morse, 1978). After returning to the swarm the scout performs a waggle dance if she has rated the site of sufficient quality to be considered. Dance followers use the information encoded in the dance to locate the advertised site, which they then independently evaluate for quality.

The number of dance circuits in the first dance performed by a returning scout is positively correlated with the scout's perception of the site's quality. After completing her dance, the scout leaves the swarm to re-evaluate the nest site before returning again and dancing another time for the same site. Each time an individual scout dances for the same nest site after having re-evaluated that site, she reduces the number of dance circuits by a fixed number of waggle runs (approx 17 dance circuits according to Seeley and Visscher 2008), regardless of the site's quality (Seeley, 2003). This means that high quality sites are advertised for longer than poor quality sites because the initial number of circuits is higher. Thus, over time more individuals are recruited to high quality sites compared to sites of lower quality.

While inspecting a potential nest site, a scout estimates the number of other scouts that are also evaluating the site. If this number exceeds a threshold ("quorum") the scout returns to the swarm and signals that the quorum has been reached by "piping", an auditory signal produced by wing vibration (Seeley and Visscher, 2003). This piping signal informs other swarm members to prepare for flight as a decision on the new site has been made (Visscher and Seeley, 2007). Finally, when the swarm is prepared to travel to its new nest site, scouts from the chosen nest site run excitedly through the swarm, breaking up its structure and inducing other bees to take off. Although the process of swarm guidance is not completely understood, it is thought that the scouts guide the swarm by flying rapidly through the swarm in the direction of the nest site (Beekman et al., 2006; Janson et al., 2005; Latty et al., 2009; Schultz et al., 2008). Upon arrival at the new site, a new colony is established.

Depending on the species, bees have different requirements for their nest sites. *A. mellifera* is a cavity-nesting species. It has very specific nest-site requirements with respect to the volume of the cavity and size of its entrance (see above). Open-nesting species, such as the red dwarf honeybee *A. florea* have less specific requirements, as they only require a shaded location, usually a twig or branch, from which to suspend their single comb (Wongsiri et al., 1996).

Differences in nesting requirements between *A. mellifera* and *A. florea* have an impact on both the frequency with which they move to a new home as well as the accuracy of the decision-making process employed. The cavity nesting *A. mellifera* will, in general, only select a new nest site during its reproductive cycle. Its selection process is very precise, and a swarm will only lift off when a decision for a single nest site has been made. This is reflected in the dances on the swarm prior to lift-off; approximately 30 minutes before lift-off the dances generally converge on the chosen site (Seeley and Buhrman, 1999). A swarm that is forced to lift off prior to such convergence will settle again and restart the selection process (personal communication, Madeleine Beekman).

In contrast, *A. florea* does not migrate solely for reproductive purposes. Colonies of *A. florea* also migrate to follow seasonal changes in food abundance (Wongsiri et al., 1996) and in response to changes in nest cover (Seeley et al., 1982). Unlike the dances of *A. mellifera*, the waggle dances performed by *A. florea* scouts are very imprecise (Beekman et al., 2008). *A. florea* scouts do not usually leave the swarm between bouts of dancing, hence they do not continually re-evaluate a nest site in the same way as *A. mellifera* does (Makinson et al., 2011). This makes it very unlikely that *A. florea* uses location-based quorum sensing to determine the time for lift-off. Although *A. florea* scouts will ultimately stop dancing for a site, there is no evidence for site quality dependent dance attrition in *A. florea* (Makinson et al., 2011). Further, it is not clear to what extent (if at all) site quality influences the duration of the waggle dance in *A. florea*, while it has a significant impact on *A. mellifera*'s dance behaviour. In *A. florea* often many different locations are advertised at a given time during the nest-site selection process. It is still unclear exactly what *A. florea* scouts advertise, but it seems that they indicate a general directional preference within their dances rather than an exact location (Makinson et al., 2011).

6. How habitat shapes choice: Decentralized decision making in spatial environments

Nest-site selection constitutes a decentralized decision-making process which can be observed in many social insects (Visscher, 2007). During selection a social insect colony faces the problem of finding a new home that suits its requirements. The underlying mechanism must provide a means by which different nest-sites can be evaluated and compared, so that the colony is able to choose the best option given a set of prospective sites in a dynamic environment. As a colony might compare different sites of similar quality, the decision-making process needs to converge, in order to guarantee that a colony ends up with a decision on a single site, rather than getting stuck between multiple solutions. Another problem the nest-site selection process has to tackle is the speed-accuracy trade off, which means that although a colony needs to decide on the best nest-site available, it needs to do this within a feasible amount of time, as it is usually vulnerable to the environment during the selection process. The faster the selection process the higher the chance to end up in a suboptimal solution. On the other hand, the longer the decision-making process takes, the longer a colony will be exposed to the environment.

As pointed out in the previous chapter the nest-site selection process of honeybees varies across honeybee species. While the European honeybee *A. mellifera* exhibits a very precise decision-making process (a swarm will only lift-off when a decision on a nest-site has been made), other species such as the Asian dwarf honeybee *A. florea* show a more fuzzy selection process.

The main distinction between those two species in terms of housing are their nesting requirements. As a cavity nesting species, *A. mellifera* requires a potential future nest-site to fulfil several particular requirements, while *A. florea* literally only requires a shady tree branch upon which to build its nest. The nesting requirements of a species will affect its choice during the selection process. While *A. mellifera* has to decide between a handful of nest-sites which are usually located far away from each other, potential nest-sites are ubiquitous in the case of *A. florea*.

An aspect which has not yet been covered in theoretical investigations on honeybee nest-site selection is to what extent the selection process has been shaped by the environment it operates in. In other words, are differences in nest-site selection which can be observed between cavity and non-cavity nesting species a result of the species' natural habitats? To investigate this question we use an extended version of the individual-based nest-site selection model for honeybees developed by JANSON ET AL. (2007). While JANSON ET AL.'s model is purely probabilistic (distances to nest-sites are encoded in different probabilities of finding them), we embedded the nest-site selection process in a spatial environment. Accordingly, the bees in the model were equipped with a scouting and flying behaviour that enabled them to sample the environment and find suitable nest locations that would be considered during decision making.

6.1. Related work: Models of honeybee nest-site selection

Several models have been used to investigate various aspects of the nest-site selection process in honeybees. BRITTON ET AL. (2002) modelled the information spread regarding potential nest-sites among bees using a differential equation model. In particular they compared what effect the possibility of direct and indirect switching between nest-sites has on the decision-making process. Direct switching implies that bees that abandon a given nest-site can directly switch to an alternative without considering the abandoned nest-site again. In the case of indirect switching bees will fall into a state where all nest options are considered. BRITTON ET AL. were able to show that under the assumption of direct switching bees are always able to choose the best option, whereas a colony might settle on a suboptimal choice assuming indirect switching if the better option is discovered too late in the decision-making process. BRITTON ET AL. were also able to show that the nest-site selection process does not require a comparison between nest-sites on an individual level, but that the information spread and thus recruitment within a colony will lead to decision at the colony level.

MYERSCOUGH (2003) studied the impact of recruitment dances on the selection process. Using a Leslie matrix model for population growth, she was able to show that dance attrition (i.e., the decrease in waggle dances for a site over consecutive visits) and the associated decline in recruitment over time for a nest-site regardless of its quality play a crucial role in the selection process. This is due to the fact that the quality of sites does not change over consecutive visits, in contrast to foraging where an advertised resource diminishes due to the bees' exploitation over time. Dance attrition provides such a decrease and thus allows the incorporation of newly found nest-sites into the selection process. Additionally, it prevents a swarm from becoming deadlocked in a decision between two nest-sites of equal quality.

PASSINO AND SEELEY (2006) developed an individual-based model to study the speed accuracy trade-off in honeybees' nest-site selection. Given a landscape of nest-sites, where each nest-site had the same likelihood at being discovered, PASSINO AND SEELEY investigated three aspects which govern the speed accuracy trade-off during nest-site selection: quorum size, dance attrition and recruitment rate. Quorum size determines how many bees are necessary for a decision to be made. Dance attrition controls how long individuals will promote individual nest-sites and thus how much time a site has to enter the selection process (by being successfully promoted) before it is forgotten, and the recruitment rate steers the exploitation-exploration ratio of a colony (i.e., how many individuals scout for new nest-sites and how many individuals evaluate known nest-sites). Using a Monte Carlo simulation PASSINO AND SEELEY explored the parameter-space of these three aspects. Their findings suggest that evolution tuned quorum size and dance attrition in bees in such a way that speed and accuracy are balanced. In addition they were able to show that an adaptive recruitment rate (i.e., the probability to start scouting for new nest-sites negatively correlates with the number of dances on a swarm) which can be observed during nest-site selection helps to foster a decision in situations where good nest-sites have been found (thus many dances occur on the swarm) and also prolongs the exploration in situations of poor choice.

PERDRIAU AND MYERSCOUGH (2007) developed a density-dependent Markov process model to study the impact of noise on nest-site selection process. Noise can enter the selection process via random events on various levels. One example are aberrations in individuals' quality assessments, which can lead to nest-site promotion behaviour that does not correspond to the actual quality of the site. Another factor that can introduce noise in the selection process is the delayed discovery of sites, as it can steer a swarm out of a nearly made decision or prolong the whole process. PERDRIAU AND MYERSCOUGH's results suggest that event-induced noise does indeed influence a swarm's ability to make the right decision and that the noise level increases with increasing quality of the sites present in the selection process.

JANSON ET AL. (2007) used an individual-based nest-site selection model to study the impact of scouting behaviour and adaptive recruitment on the nest-site selection process. They were able to show that a recruitment strategy like that which is found in bees' foraging behaviour leads to a good balance between exploitation and exploration. Similar to PASSINO AND SEELEY (2006), the probability of an idle bee to start scouting increases with the length of time it is unable to locate a dance on the swarm to follow. In contrast to previous models, JANSON ET AL. incorporated the distance of a nest-site to the swarm in the probability of finding the site. This allowed them to investigate to what extent different search strategies and the associated difference in discovery time impact the nest-site selection process. Three search strategies were investigated: a uniform search strategy (each site is discovered with equal likelihood regardless of its distance to the swarm), a distance search strategy (the probability to discover a nest-site decreases with its distance to the swarm) and distance-squared search strategy (the probability of finding a nest-site is inversely proportional to the square of the distance to the swarm, which takes into account that the search space increases quadratically the further away you move from the swarm). Their findings suggest that the quality of a scouting strategy depends on the travelling costs. Under a uniform search strategy a colony will always choose the better nest-site and disregard the distance to the nest-site. However, given two nest-sites of the same quality and different distances, the closer nest-site should be chosen as it minimizes the distance the whole swarm has to travel and thus decreases the time a swarm is exposed to the environment. Using the distance or distance-squared search strategy a swarm is able to discriminate between those two sites and will choose the site which is closer. JANSON ET AL. also tested a scenario with nest-sites of different qualities and different distances to the swarm. In a situation where the better nest-site is further away from the swarm than a suboptimal choice, the bees' should select for the site which is further away, as it imposes more costs to move into a lower quality site than it does to move further. While bees show this behaviour using the uniform and distance search strategies, their ability to choose the superior site decreases with increasing distance under the distance squared-search strategy.

MARSHALL ET AL. (2009) set the nest-site selection behaviour of honeybees and ants in analogy with the decision-making found in primate brains. MARSHALL ET AL. argued that like in the brain, where a neuron fires once an activation threshold is reached, social insects will recruit individuals for different choices in different opposing "choice populations", with a decision being made once one of the populations reaches a certain size and thus quorum is reached. Using BRITTON ET AL. (2002)'s nest-site selection model they investigated to which extent the honeybee nest-site selection behaviour approximates a statistically optimal decision-making process, which can be observed in 'diffusion models' of primate decision making.

Recently NEVAI ET AL. (2010) proposed two compartment models to study the stability of choice during the nest-site selection process. Their first model simulates a swarm's assessment process of a single site, which constitutes the decision of whether a site is of sufficient quality to enter the selection process. Two equilibrium states of the assessment process model are investigated: the disinterested equilibrium, where bees will disregard a site; and interested equilibrium, where bees will show interest in a given site. The stability of these equilibria is investigated regarding the basic and absolute recruitment numbers, which denote the sensitivity of the swarm towards individual recruitment dances. Additionally, NEVAI ET AL. estimated a critical site quality threshold. If the quality of a site exceeds the threshold it enter the selection process (by being promoted), otherwise it won't. Using their second model NEVAI ET AL. studied a swarm's discrimination process between two nest-sites which differ in both quality and the time at which they are discovered by the swarm, and the disinterested and interested equilibria for the decision-making process were estimated regarding the basic recruitment numbers.

6.2. Model for nest-site selection in a spatial environment

The individual-based model which was used here to simulate the nest-site selection process of *A. mellifera* under different spatial setups is an extended and revised version of Janson et al.'s nest-site selection model (Janson et al., 2007). The model does not cover the behaviour of the reproductive swarm as a whole, but simulates bees that are involved in the selection process, which are estimated to be $2-5\%$ of the swarm (Seeley and Buhrman, 1999). In the following we will introduce the model and outline the differences to Janson et al.'s model.

The model operates in discrete time-steps representing 1 second of real time. Note that in the original model (Janson et al., 2007), a time-step size of 6 seconds was used. As we are interested in modelling nest-site selection within a spatial environment, which involves simulating the flight and scouting behaviour of bees, such a temporal resolution would be too coarse. Scouting honeybees can travel with a maximum speed of approximately 5 meters per second (Beekman et al., 2006), meaning that they can travel a maximum distance of 30 meters in a time-step of 6 seconds. In a spatial simulation this would make it very likely for a bee to miss a potential nest-site by simply flying over it, thus a smaller timescale had to be used.

During each simulation-step all virtual bees are invoked in random order. Each bee will act according its current behavioural state. In nature the duration of behaviours such as scouting, missing and nest-site assessment can vary. To account for this, each of these behavioural states E are associated with a mean duration time T_E. The exact duration of the respective behaviour is determined by $T(E) = \lambda \cdot T_E$, where $\lambda = \mu/10$ is a scalar factor, with μ being drawn from a chi-square distribution $\chi^2(10)$ [1]. In the following the mean duration times for time-varying behavioural states E will be stated.

6.3. Behavioural states

The following section outlines the behavioural states used in the nest-site selection model. Each bee can be in one of eight possible states. The states are briefly outlined in Table 6.1, a state diagram which depicts the possible transitions between the states is provided in Figure 6.1.

Resting

Not all of the bees that are involved in the nest-site selection process are active participants all the time. Several empirical studies (Beekman et al., 2007; Camazine et al., 1999)

---

[1] Note that this leads to an expected value of 1 for λ

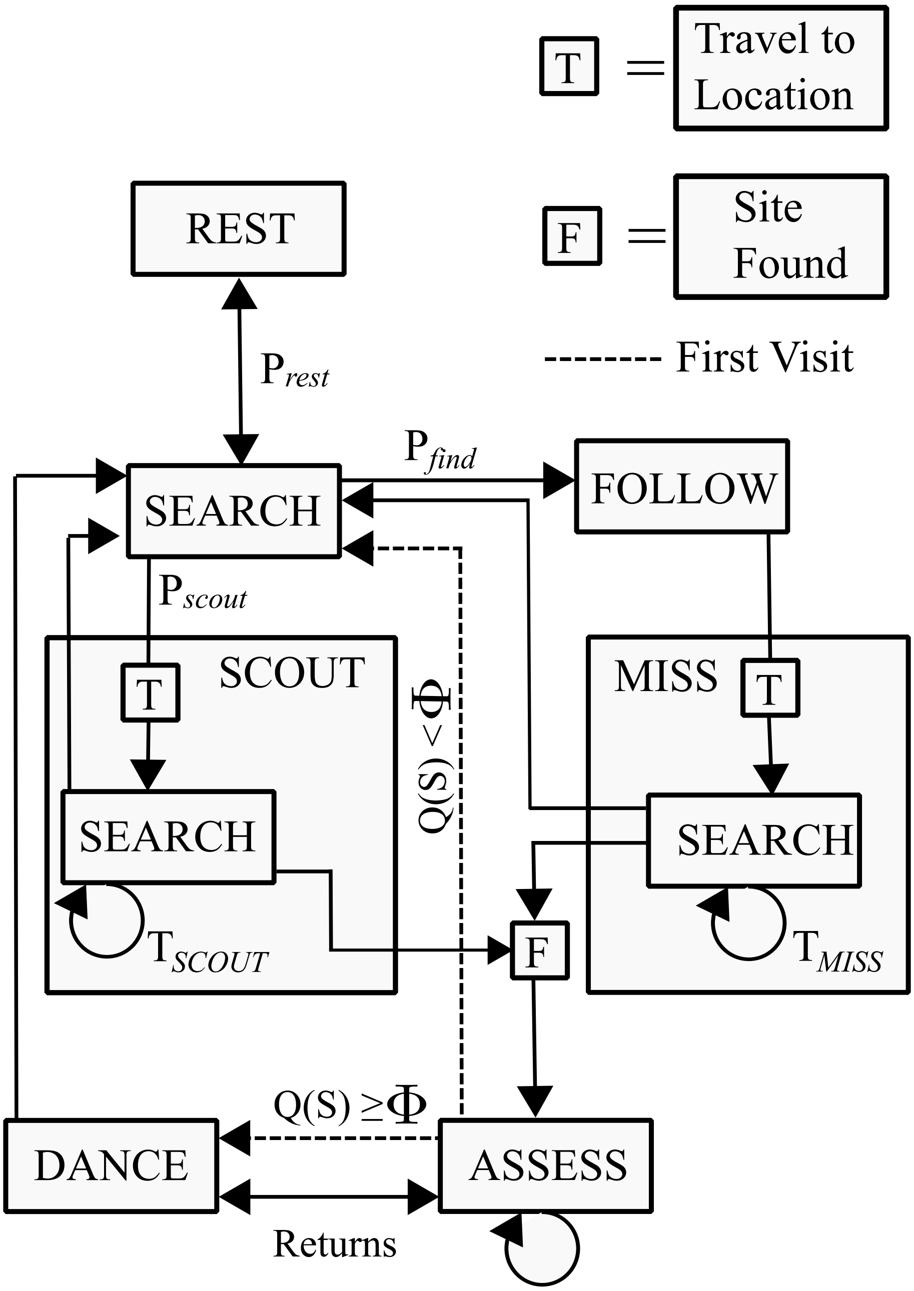


Figure 6.1.: State diagram showing the individual behaviours underlying the honeybee nest-site selection process. Details of the states and state transitions are outlined in Section 6.3.

State	Description
REST	The bee is resting on the swarm and currently not involved in the selection process
SEARCH	The bee is on the swarm and searching for a dance to follow
SCOUT	The bee searches the surroundings for potential nest-sites
ASSESS	The bee assesses the quality of a potential nest-site
DANCE	The bee advertises a nest-site on the swarm by dancing
FOLLOW	The bee is on the swarm and follows a dance it found during SEARCH
TRAVEL	The bee flies towards a destination (e.g., nest-site, swarm location)
MISS	The bee misread the dance and searches around the wrong location for the advertised nest-site

Table 6.1.: Behavioural states of the nest-site selection model.

have noted that bees tend to drop in and out of the selection process, by switching from their resting state into searching for a dance and vice versa. In accordance with these studies, the probability that a resting bee will start to search for a dance or start resting was set to $P_{rest} = 0.002$. This leads to an expected switch interval of 500 seconds between resting and searching, if potential switches from SEARCH to other behavioural states (SCOUT, FOLLOW) are disregarded.

Searching

A bee that is in the behavioural state SEARCH will try to locate a dance for a nest-site on the swarm. The probability that it will be able to locate a dance increases with the number of dances D which are currently being performed on the swarm. In accordance with Tautz and Rohrseitz (1998) the probability to find a dance was set to $P_{find} = 0.005 \cdot D$. If the bee is able to find a dance it will randomly choose a dance from those that are currently being performed on the dance floor. A bee will only follow the chosen dance if it has less than 7 followers. The probability that it will start to follow a dance is given by $P_{follow} = 0.2^{min\{2,f\}}$, where f denotes the number of bees already following the dance. If a bee chooses not to follow the dance it will remain in the SEARCH state.

The longer a bee is unable to locate and follow a dance, the more likely it gets that the bee will start to scout. The probability that a searching bee will switch to scouting depends on the time t it has been searching for a dance (i.e., has been in state SEARCH) and is given by

$$P_{scout}(t) = \frac{t^2}{t^2 + \theta^2} \tag{6.1}$$

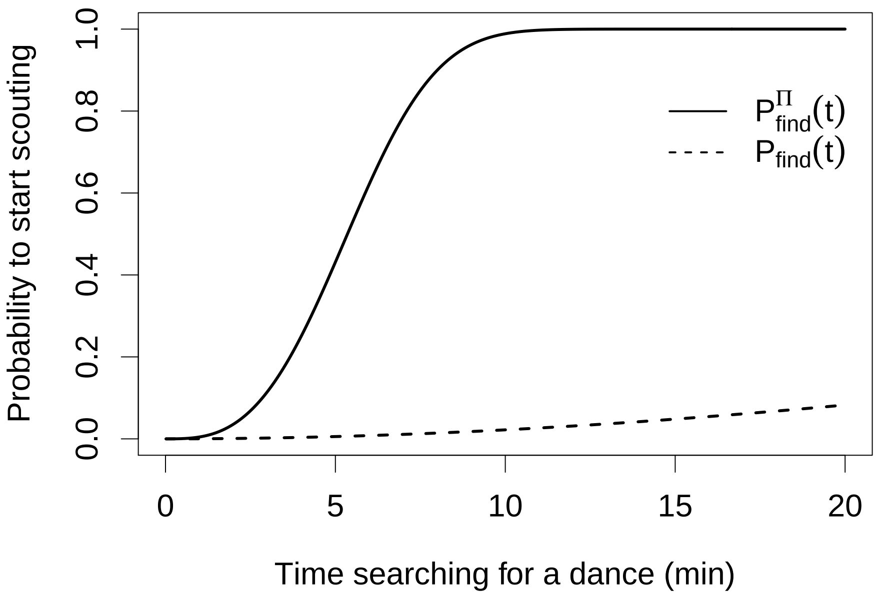


Figure 6.2.: Evolution of the cumulative ($P^{\Pi}_{scout}(t)$) and per time-step ($P_{scout}(t)$) probability that a bee in the state SEARCH will start scouting.

where t denotes the number of time steps it has been searching for a dance and $\theta = 4000$ denotes a threshold. The reader should note that $P_{scout}(t)$ corresponds to a response function which is used in response threshold models, which were discussed earlier (see Chapter 2). As $P_{scout}(t)$ is applied in the case of a searching bee at each time-step, the cumulative probability that a bee will start to scout after t seconds of unsuccessful dance searching is given by:

$$P^{\Pi}_{scout}(t) = 1 - \prod_{i=0}^{t}(1 - P_{scout}(i)) \tag{6.2}$$

Figure 6.2 depicts the evolution of $P_{scout}(t)$ and $P^{\Pi}_{scout}(t)$ over 20 minutes. As can be seen, a majority of bees will have left the swarm after around 10 minutes of searching for a dance to proactively look for a suitable nest-site themselves.

The combination of P_{find}, P_{follow} and P_{scout} modulates the exploration/exploitation rate of a swarm. If very few potential nest-sites have been found and thus the number of dancing bees is low, bees searching for a dance are likely to become scouts. When many sites have been found and therefore dances are abundant, a searching bee is likely to find a dance to follow and will become a recruit instead of a scout.

Scouting

As LINDAUER (1955) observed, bees usually scout the area surrounding the swarm for about 20 minutes before returning. Accordingly, a bee will scout for $T_{SCOUT} = 1200$ simulation steps. While scouting the virtual bees move through a 2-dimensional environment in search of potential nest-sites. This is a major difference to JANSON ET AL.'s model where scouting was modelled stochastically, and also affects the behaviour of the model. In the original model a scouting bee will always scout for T_{SCOUT} iterations before either finding a nest-site or returning to the swarm, meaning that it will take at least T_{SCOUT} iterations to find a nest-site. This is not the case here; bees will sample the 2-dimensional environment and if they find a nest-site they will immediately stop scouting.

In order to guarantee that a scout that is searching the 2-dimensional environment returns to the swarm after T_{SCOUT} simulation steps, the scouting behaviour was split into two sub-behaviours

1. scouting: a bee will scout as long as it is able to be back at the swarm after T_{scout} time steps.
2. returning: if the remaining scouting time is smaller or equal to the time needed to return to the swarm, a scout returns to the swarm.

Spotting a potential nest-site In nature a bee can spot a target if the target subtends the bee's visual angle α_{min} (see Figure 6.3 for a sketch) which can range between five and fifteen degrees (GIURFA ET AL., 1996; KUGLER, 1933). The diameter of nest boxes normally used in nest-site selection experiments is around 40cm. Here we assume a minimal angle of $\alpha_{min} = 8$ degrees, which means that a scout can spot a nest-site up to a distance of approximately 280cm. After a successful discovery, a scout will immediately start to assess the site and thus change its state.

Scouting strategy: The exact way scouts search the environment is still unknown. Recent studies (REYNOLDS ET AL., 2007, 2009) suggest that bees exhibit a Lévy flight/walk when searching for resources and misplaced food sources. The Lévy walk is a random walk which is characterized by the fact that the length of the movement segments without directional alternation is distributed according to a power law distribution. Besides bees, several other animals such as albatrosses (VISWANATHAN ET AL., 1996) and deer (FOCARDI ET AL., 2009) have been suggested to exhibit Lévy walk patterns when searching for resources. Whether or not animals really exhibit Lévy flights is still debated (BENHAMOU, 2007; REYNOLDS, 2008). One of the main arguments against the Lévy flight is that for an animal to really exhibit this search strategy the scale-free distribution of the lengths of its movement paths has to be an intrinsic property of the animal. If alternations in

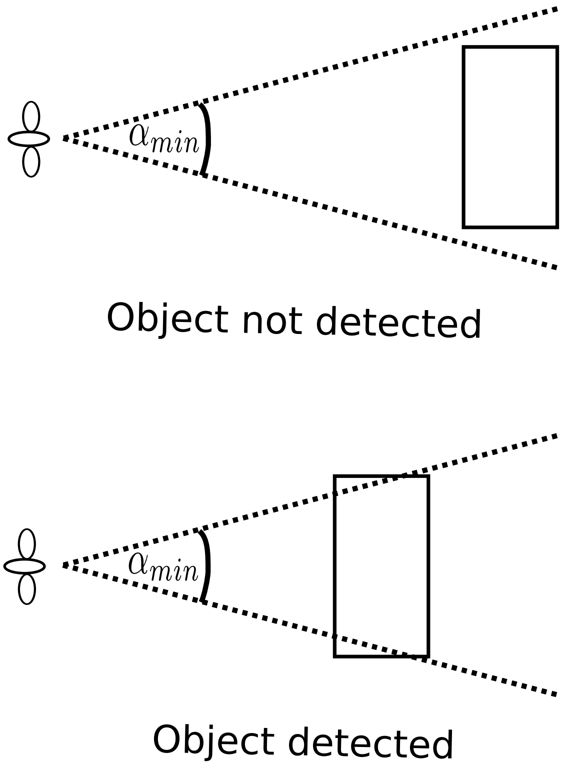


Figure 6.3.: Sketch of a bee's ability to spot a target. A bee can detect an object if the object subtends the bee's visual angle α_{min}.

the travel direction are caused by the environment (e.g., visual, chemical cues), then what might appear to be a Lévy walk is in fact none, as it is caused by the environment, and a different environment might cause different (non-Lévy) travel paths of the animals.

In the case of honeybees trying to relocate food sources (Reynolds et al., 2007, 2009), it is unclear if the Lévy flight trajectories observed in experiments are an intrinsic feature of the bees' search movement or caused by external factors such as environmental cues or the bees' memory. In the experiments, the bees knew what they were searching for (i.e., a foodsource they had visited before which was no longer where they expected). As pointed out in Chapter 5 honeybees are able to incorporate lots of private information in their foraging routine, which could thus shape their flight trajectory when trying to relocate known resources.

When searching for a new nest-site, scouting bees are not able to fall back on private information, they need to locate potential nest-sites that are unknown to them. To simulate their scouting strategy an intermittent search strategy was used in this model. An intermittent search consists of two phases, a phase of fast motion during which individuals traverse the environment and a phase of slow motion which allows the detection of resources. Intermittent search strategies have been shown to be particularly useful when searching for hidden unknown targets (Bénichou et al., 2005) and are thought to be widespread in many animals (Bell, 1990; Kramer and McLaughlin, 2001).

Janson et al. (2007) suggested that an intermittent search strategy could underlie the scouting behaviour of honeybees during nest-site selection, where scouts will fly towards an area which they then search thoroughly. In our model an intermittent search strategy is realized as follows: scouts choose a random location within a search area that is defined by the range of locations that are reachable within one third of its available scouting time T_{scout}. During the travel-phase a scout will fly to the chosen location (flight behaviour is explained in the next section). After reaching the chosen location a scout will perform a correlated random walk (CRW) to search for a potential nest-site (Bartumeus et al., 2005; Zollner and Lima, 1999). Various species such as ants, beetles and butterflies have been shown to perform CRWs (Crist et al., 1992). As a result, CRW has been used to reproduce movement patterns from various experimental data (e.g., Bergman et al. 2000; Crone and Schultz 2008). In contrast to a pure random walk (i.e., Brownian motion), CRW incorporates directional persistence in movement patterns. Given a position and a direction, directional persistence can be achieved by limiting the angular displacement of the direction between successive steps.

For the scouts' movement a CRW with a fixed movement length of 1 meter per step is used. Angular displacement is achieved by adding directional noise which is drawn from a wrapped Cauchy distribution (Baschelet, 1981). Wrapped Cauchy distributions contain a shape parameter $0 \leq \rho \leq 1$ which controls directional persistence. If $\rho = 0$ the resulting walk is uncorrelated. In contrast $\rho = 1$ results in total correlation, which means that no noise is added to the direction. For the simulation runs a correlation parameter value $\rho = 0.5$ is used, resulting in intermediately correlated movement steps. Figure 6.4 depicts a sample flight path of a scout.

Flying towards a destination

A scout flying towards a destination travels with a speed of 5 meters per second. If the distance to the destination is smaller than 5 meters, the bee is placed on the destination, otherwise it will travel 5 meters in the direction of the destination. In order to prevent bees from flying in straight lines, angular noise was added from a uniform random distribution η_{fly} $(-22.5^\circ \leq \eta \leq 22.5^\circ)$. Because a bee aligns its flying direction each time step, it is guaranteed to arrive at the destination.

Site assessment

Should a scout successfully locate a potential nest-site it will assess it. Nest-site assessment in real bees usually lasts for about 10 minutes (Lindauer, 1955) which corresponds to

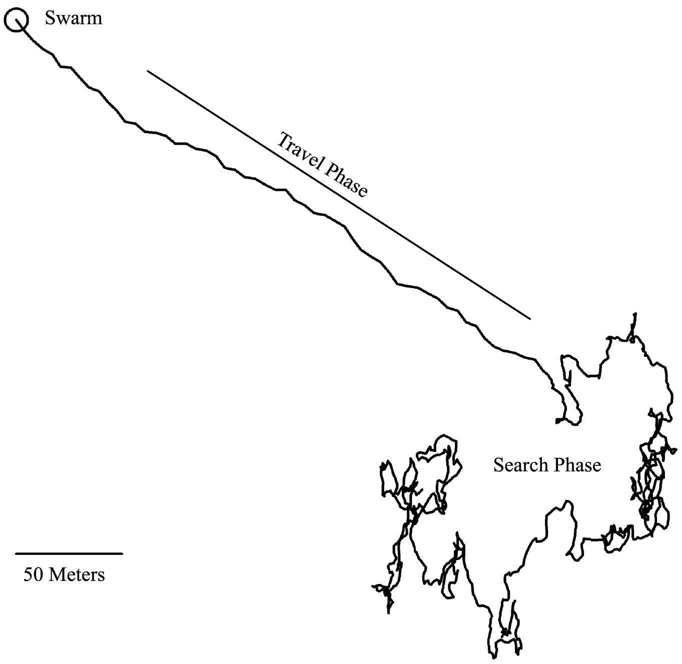


Figure 6.4.: Sample scout flight-path of a scout using an intermittent search strategy.

$T_{ASSESS} = 600$ iterations of the simulation. Each nest-site S has a certain quality Q_S, which in the simulation corresponds to a natural number in the range $[0-100]$. The quality of a nest-site S is perceived by a bee during the assessment. Quality is always perceived with some noise, thus $Q(S) = Q_S + \delta$, with δ drawn from a normal distribution $N(0, \sigma^2)$ with a standard deviation of $\sigma = 10$. If the perceived quality $Q(S)$ exceeds a bee's quality threshold Φ, the bee dances for the nest-site when it returns home. Otherwise it switches to searching after it returns home. As in the original model, the threshold Φ is set to 50 for all individuals in the simulation. After a bee has completed the assessment of a nest-site it flies back to the swarm.

Dancing

If a bee discovers a potential nest-site S (i.e., $Q(S) > \Phi$), it dances for it after returning to the swarm. The number of waggle runs performed during a dance depends on two factors, the perceived quality of the nest-site $Q(S)$ and the number of consecutive visits to the nest-site. Based on empirical data (Seeley, 2003), the simulated bees perform $Q(S)$ waggle runs after their first visit to the nest-site and $Q(S) - 16(k-1)$ after returning for the kth time. If $Q(S) - 16(k-1) \leq 0$ it will stop dancing for this site and switch to searching.

The distance to and direction of the potential nest-site are incorporated into the dance by assuming that a waggle phase lasts 2.4 seconds per kilometre and 1.5 seconds are added for the return phase (GARDNER ET AL., 2008). Thus a single dance for a potential nest-site located 1000 meters away from the swarm takes 3.9 seconds.

Following

A searching bee that has found a dance and was able to follow it, follows the dance until the dancer ceases dancing. If the bee had previously visited the advertised site, it will find that site again. The number of waggle runs a bee is able to follow determines if the follower will be able to find the advertised site. MAUTZ (1971) monitored the success rate of bees to find a given resource regarding the followed waggle dances. JANSON ET AL. (2007) used these rates to estimate the probability of a bee correctly locating the advertised nest-site after following w waggle runs

$$P_{findSite}(w) = \frac{s(w)}{1.5 \cdot u(w) + s(w)}$$

Here $u(w) = 1 - 1/\sqrt{(w+1)}$ represents MAUTZ's distribution of bees that followed w dances-cycles and were not able to locate the resource and $s(w) = w^2/(w^2 + \theta)$, with $\theta = 60$, represents the distribution of bees that were able to locate a given resource after following w waggle runs.

Successfully recruited to nest-site

If a bee has been successfully recruited for a potential nest-site and correctly read the dance it followed, it flies towards the proposed nest-site. After reaching the nest-site it starts to assess it for T_{ASSESS} simulation steps. If the assessment is successful (i.e., $Q(S) > \Phi$), the bee returns to the swarm and starts to dance for the nest-site. Otherwise, it returns to the swarm to search for new dances.

Missing the advertised nest-site

The reason why bees miss a nest-site after following a waggle dance is due to imprecision in the dance which yields variance in the transmitted directionality of the promoted site (WEIDENMÜLLER AND SEELEY, 1999). In our model this is implemented by adding noise drawn from a uniform distribution (maximum of $\alpha_{miss} = [-5^\circ - 5^\circ]$) to the direction of the advertised nest-site. After the dance has ceased, a wrongly informed bee will fly towards this wrong nest-site location. After reaching the wrong location and finding no potential site, the bee will start to search the surroundings for $T_{MISS} = 400$ simulation steps, using a CRW as in the search-phase during scouting. If a bee is able to find a nest-site (either the one it tried to locate initially or a different one) it will continue to assess the found nest-site. Otherwise it will return back to the swarm.

6.4. Experiments

6.4.1. Experimental setup

Unless stated otherwise we used the parameter values mentioned in the last section. We present the results as average values obtained from 100 independent runs. The number of individuals used in the experiments was set to $n = 500$, which corresponds to approximately the number of bees involved in nest-site selection in real honeybees. Simulations were run for over 57600 simulation steps. With each simulation step corresponding to one second in real time this corresponds to a simulation duration of 16 hours.

6.4.2. Nest-site discovery in sparse spatial environments

In the first experiment a swarm's ability to locate and decide upon nest-sites in a sparse environment under an intermittent search strategy is investigated. In nature, nest-site availability depends on the environment as well as on the requirements of the species. As pointed out before *A. mellifera* has very strict nest-site requirements and thus usually has only a few options which may be highly distributed in the environment. This can lead to a dispersal of honeybee colonies up to several kilometres (Camazine et al., 1999; Schneider, 1995). While colony dispersal is the norm in migration of *A. mellifera*, it should be noted that a few cases of colony aggregation have been reported (Baum et al., 2005; Oldroyd et al., 1995), which are attributed to nest-site richness in the respective environments as well as other factors.

To test a swarm's discovery and selection potential in a sparse environment, the swarm was placed in the center of a square area of size 100 km^2 in which 99 nest-sites were randomly distributed (see Figure 6.5 for a setup sketch). Each nest-site had to be at least 150 metres away from any other nest-site of the swarms location. An equal amount of good ($Q(S) = 70$), mediocre ($Q(S) = 50$) and bad ($Q(S) = 45$) quality nest-sites were distributed in the environment, leading to a total of 33 nest-sites of each quality. As nest-sites are uniformly distributed in the environment, the described setup leads to an average of 1 nest-site per km^2.

During each of the 100 simulation runs the colony was able to detect at least one nest-site of good quality ($Q(S) = 70$) and the decision-making process always converged towards a good quality nest-site, as indicated by a majority of individuals dancing and assessing the

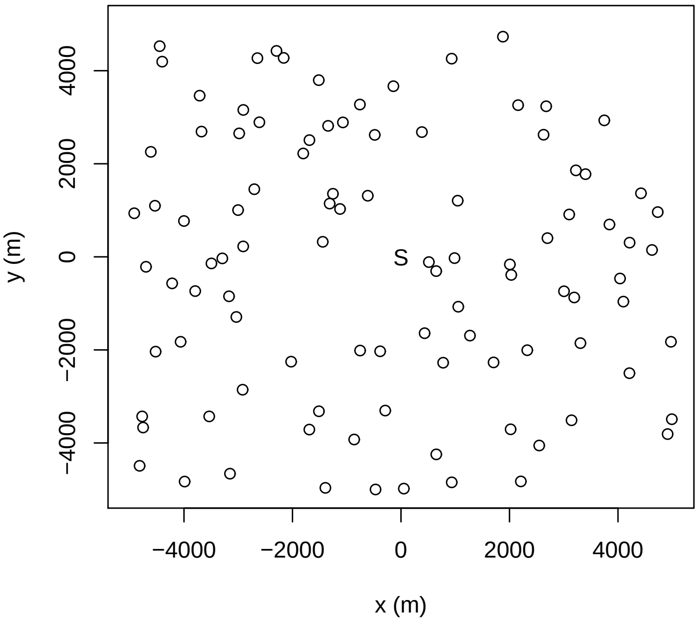
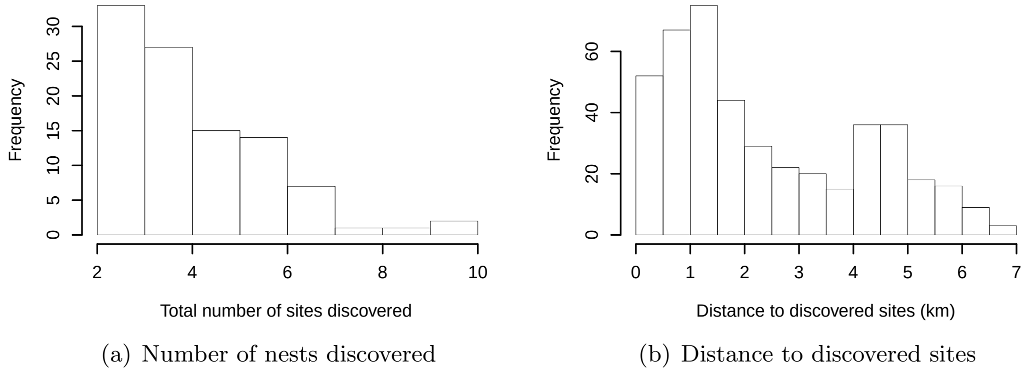


Figure 6.5.: Sparse nest-site distribution in an environment with an area of 100 km^2. *S* denotes the location of the swarm, and each *o* corresponds to a potential nest-site.

nest-site while other found alternatives were abandoned. Figure 6.6(a) depicts a histogram of the total number of nest-sites which were discovered during single simulation runs. As can be seen a colony was able to evaluate between 2 and a maximum of 10 nest-sites during the selection process. Figure 6.6(b) displays the fight distances from the swarm's position to the found nest-sites which ranged from several hundred meters up to around 7 kilometres.

As pointed out above, a colony was able to locate and select a nest-site of good quality during each simulation run, while the number of nest-sites that were discovered in total per simulation run altered. Figure 6.7(a) depicts the total number of nest-sites a swarm discovered during a simulation run regarding the discovery rank of the first nest-site of good quality which was discovered (i.e., a discovery rank of i indicates that the nest-site was the ith site discovered by the swarm).



(a) Number of nests discovered

(b) Distance to discovered sites

Figure 6.6.: Histograms of the total number of discovered nest-sites during single simulation runs (a) and the distance the distance the discovered nest-sites had to the swarms position (in km) (b).

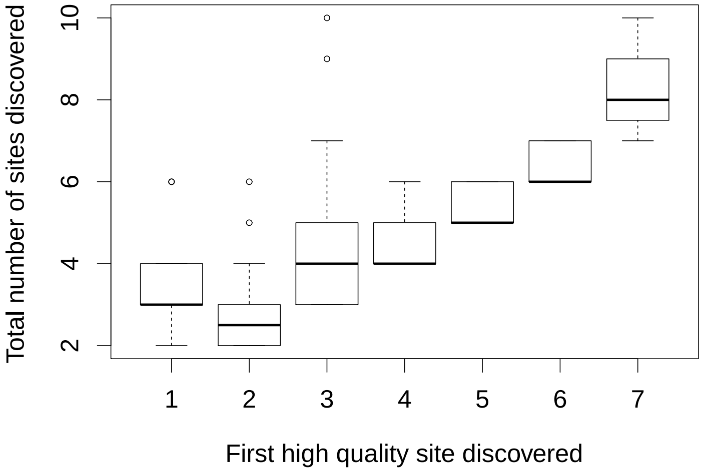


Figure 6.7.: Correlation of total number of nest-sites discovered in a single simulation run and discovery rank of first high quality nest-site.

As can be seen the total number of sites discovered during a simulation run correlates with the discovery rank of the first good quality nest-site that was found. This is due to the fact that the discovery of a site with good quality is accompanied by a quick built-up of strong recruitment towards this site, which leads to an increased number of dances and thus a decline in scouting. In cases where only sites of medium or bad quality are found,

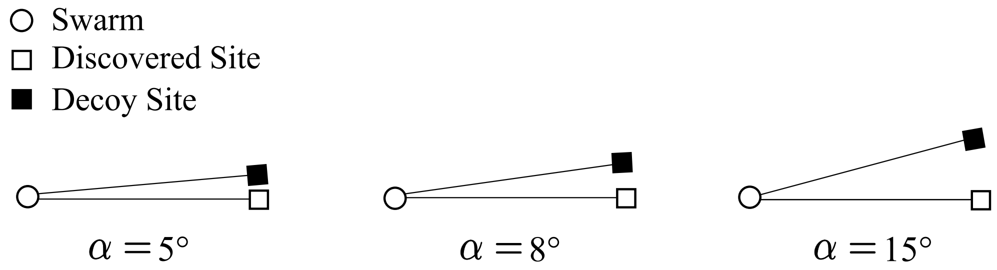


Figure 6.8.: Sketch of experimental setup to test the influence of local choice on convergence towards a single solution.

recruitment is retained and thus many individuals will decide to scout, which eventually leads to discovery of a good site.

The correlation between sites discovered in total and rank of the first good site discovered is a good depiction of the ability of the nest-site selection process to modulate between speed and accuracy during the selection process given its current set of choices. This corroborates previous findings on the adaptivity of the decision-making speed during nest-site selection in social insects (Passino and Seeley, 2006; Sumpter and Pratt, 2009). Additionally these results suggest that an intermittent search strategy as used in our model is indeed a valid search strategy in sparse spatial environments, as it enables a swarm to find a sufficient amount of nest-sites in a wide spatial range. Again this agrees with observations from experimental data on nest-site selection in honeybees (Lindauer, 1955).

6.4.3. Influence of increased local choice on site convergence

As we have seen in the last section the nest-site selection process is able to locate and select good nest choices in sparse environments, a scenario a migrating *A. mellifera* swarm is likely to face. As pointed out earlier other bee species such as *A. florea* face an abundant set of choices during migration and also exhibit a selection process that differs from the one found in *A. mellifera* in terms of dance-precision and consensus before lift off (Beekman et al., 2008; Makinson et al., 2011).

While it was assumed in the previous model (Janson et al., 2007) that a bee would unsuccessfully scout for around 7 minutes before returning to the swarm, here the bee will actually search its surrounding for the missed site to assess it. Search behaviour after missing an advertised site has been reported for relocated food sources (Reynolds et al., 2009) and is thus likely to occur when missing a nest-site. In such a case the bee has a chance to discover the nest-site it missed as well as other sites that are in its vicinity. If the bee discovers a nest-site it was not looking for, this can impact the selection process. “Non-intended scouting” can lead to the discovery of nest-sites and can thus increase inter-nest-site competition and slow down the convergence process towards a nest-site.

To examine to what extent the selection process has been shaped by the environment, thus leading to a very precise selection process in *A. mellifera* and a rather fuzzy one in *A. florea*, we tested the influence of close local choice on the selection process. A swarm was situated in an environment containing two nest-sites. Both nest-sites were equidistant to the swarm (i.e., 250 metres) and had a given angular separation α between them regarding the swarm's position (see Figure 6.8 for a sketch of the experimental setup). At the start of the simulation a random bee started to dance for one of the nest options, while the other site remained undiscovered and acted as a decoy site, which could either be discovered via regular scouting or during search after missing the discovered site.

Three angular separations of $\alpha \in \{5, 8, 15\}$ degrees were tested under low, medium and good nest-site quality conditions $Q(S) \in \{45, 50, 70\}$. Figure 6.9 depicts the number of bees at the discovered nest-site as well as at the decoy nest-site for the resulting 9 scenarios.

Both quality and angular separation affect the incorporation of the decoy nest-site into the selection process. Given a small angular separation ($\alpha = 5°$) between both sites, a bee which misses the advertised nest-site is very likely to discover the decoy nest-site during the search, following a failure to locate the advertised nest-site. As can be seen in Figures 6.9 a, b and c, given a small angular separation ($\alpha = 5°$) the build up of bees at the decoy nest-site is very strong. In the case of the nest-sites being of good or mediocre quality ($Q(S) \in \{50, 70\}$) the decoy nest-site is never discovered via regular scouting but always as a result of bees missing the initially advertised nest-site. Given poor site conditions bees sometimes discover the decoy nest-site via conventional scouting. A decrease in quality further increases the rate of which bees will miss an advertised site as it leads to a reduction of dance circuits during the waggle dance resulting in an increased chance of error and thus missing the advertised site. It should be pointed out that site-discovery via missing happens in both directions (i.e, bees that were initially recruited for the decoy nest-site will end up at the original site and vice versa). This makes a convergence towards one nest-site nearly impossible in situations of small angular separation as sites are never able to leave the competition as they are constantly rediscovered.

Increased angular separation between the two nest-sites decreases a bee's probability of ending up at the decoy nest-site when missing the advertised site. While such a tendency is still observable in a situation of of an angular separation of $\alpha = 8°$ (Figures 6.9 d, e and f) it is nearly gone in the case of $\alpha = 15°$ (Figures 6.9 g, h and i). Given a large angular separation between nest-sites ($\alpha = 15°$), missing behaviour will only result in the detection of the decoy nest-site when the quality of the initially found site is mediocre or bad ($Q(S) \in \{45, 50\}$). However alternative site discovery as a result of missing happens very seldom and thus has no visible effect on the swarm's convergence in choice, in contrast to situations of smaller angular separation.

These results indicate that the nest-site selection process is indeed sensitive to the spatial distribution of nest-sites in the environment, even when the angular separation α between two sites is greater than the angular error α_{miss}. Given an environment where potential sites are too densely clustered, a selection process that is geared towards selection of a single nest-site is likely to run into difficulties, which can either delay a swarm's convergence towards a decision or cause a deadlock between decisions. A decrease in site quality further increases the obstructive nature of environments with densely clustered nesting options, as it leads to an increase in bees missing advertised sites and thus discovering close-by alternatives.

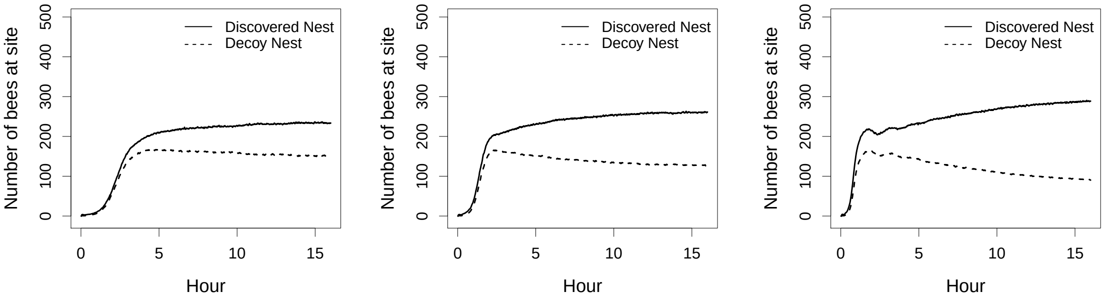
6.4.4. Influence of increased local choice on decision making

The previous experiment has shown that nest-sites which are in the close vicinity of discovered nest-sites can influence a swarm's convergence in decision, as they are likely to be discovered when bees miss a given nest-site and thus enter the selection process causing an ongoing flux of bees between the nest-sites and thus delaying, or in cases of very close and bad quality, disabling the convergence towards a single site. In this experiment we wanted to test the influence of spatial proximity when the swarm has to decide between two nest-sites.

To test a swarm's decision-making ability in the case of increased local choice, a swarm swarm was situated in an environment containing 4 nest-sites. All nest-sites were equidistant to the swarm (i.e., 250 metres). In pairs of two, the nest-sites were located in two separate regions on opposing sides of the swarm (see Figure 6.10 for a sketch of the experimental setup). At the start of the simulation the swarm discovered one nest-site in each region, while other nest-site in the region functioned as a decoy site.

A nest-site pair in region p had a given angular separation α_p regarding the swarm's location. While the angular separation between the nest-site pair in region 2 was held constant at a level which would not impact the convergence towards the site discovered initially (i.e., $\alpha_2 = 15°$), the angular separation between the nest-site pair in region 1 was altered (i.e., $\alpha_1 \in \{5°, 8°, 15°\}$). The three resulting spatial setups were tested under three different quality conditions ($Q(S) \in \{45, 50, 70\}$ corresponding to low, medium, and good quality), with each potential nest-site exhibiting the same quality.

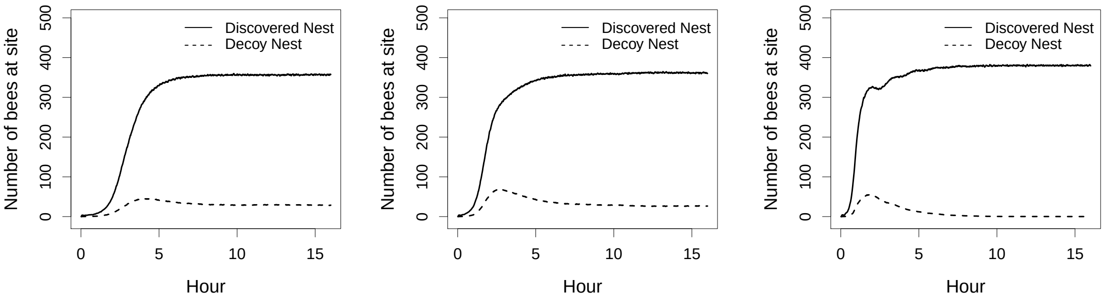
Figure 6.11 depicts boxplots of the number of bees located at each nest-site during the last time-step of the simulation (i.e., after 57600 simulation steps). As can be seen in Figures 6.11 a,b and c, regions that contain nest-sites with very close proximity to each other (i.e., $\alpha_1 = 5°$) will win the swarm's attention, leading to bees on either of the sites



(a) Separation $\alpha = 5°$
Low Quality ($Q(S) = 45$)

(b) Separation $\alpha = 5°$
Medium Quality ($Q(S) = 50$)

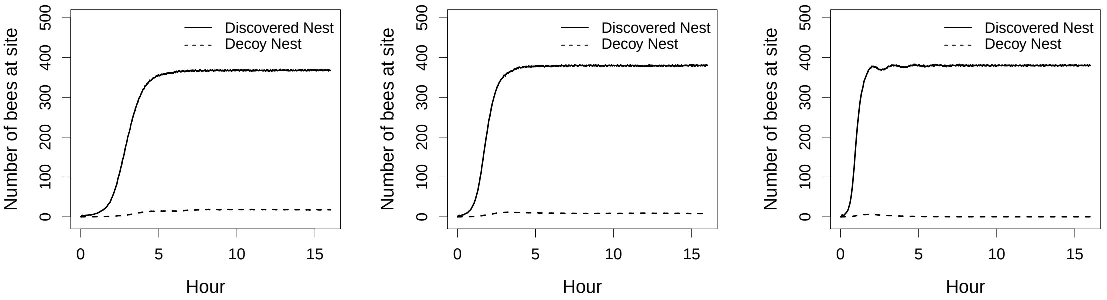
(c) Separation $\alpha = 5°$
High Quality ($Q(S) = 70$)



(d) Separation $\alpha = 8°$
Low Quality ($Q(S) = 45$)

(e) Separation $\alpha = 8°$
Medium Quality ($Q(S) = 50$)

(f) Separation $\alpha = 8°$
High Quality ($Q(S) = 70$)



(g) Separation $\alpha = 15°$
Low Quality ($Q(S) = 45$)

(h) Separation $\alpha = 15°$
Medium Quality ($Q(S) = 50$)

(i) Separation $\alpha = 15°$
High Quality ($Q(S) = 70$)

Figure 6.9.: Average number of bees assessing a nest-site over 57600 simulation steps (i.e., 16 hours) for different degrees of angular separation $\alpha \in \{5, 8, 15\}$ and site qualities $Q(S) \in \{45, 50, 70\}$.

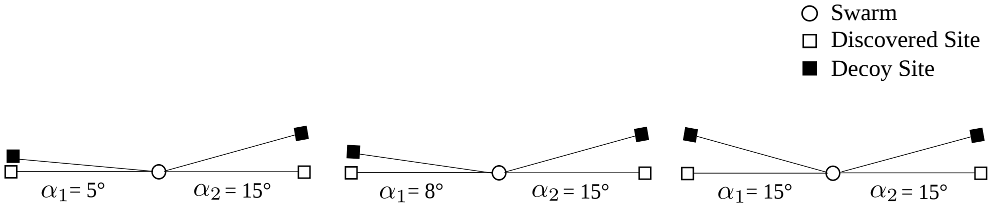


Figure 6.10.: Sketch of experimental setup to test the influence of local choice on decision-making.

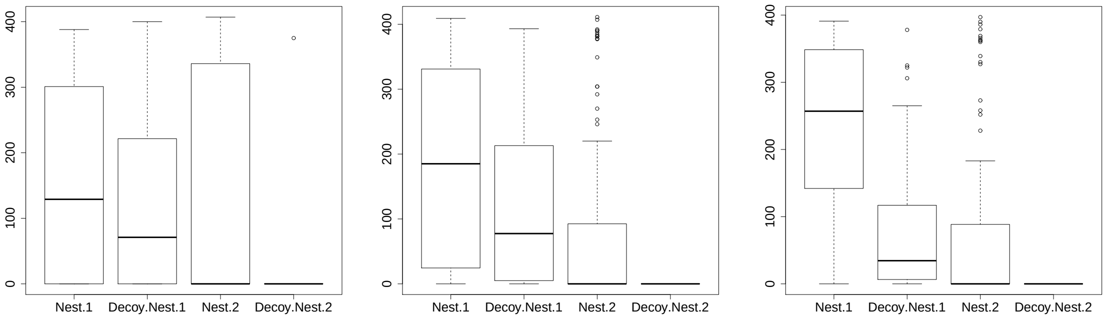
at the end of a simulation. As in the previous experiment, the swarm's ability to neglect the decoy site and converge towards a decision increases with the quality of the nest-sites. However, even in a situation of good quality (Figure 6.11 c), where the influence of the decoy nest-site is not as strong as under low or medium quality setups, the spatial distribution of the nest-sites shapes the swarm's choice as a most of the bees end up at the initially discovered nest-site in region 1 in a majority of the simulation runs. If spatial nest-site distribution did not have any effect on the selection process one would expect a uniform distribution over the 2 initially discovered sites, which is clearly not the case.

With an increase of angular separation between the nest-sites in region 1 (i.e., $\alpha_1 = 8°$), the trend towards the region with the denser nest-site distribution prevails in situations where nest-sites are of low and medium quality (Figure 6.11 d,e). In situations of high nest-site quality (Figure 6.11 f) the swarm is nearly always able to neglect the decoy nest-site in region 1 in the decision-making process, and the convergence towards nest-sites in either regions is nearly balanced.

When the angular separation between nest-sites in region 1 further increases (i.e., $\alpha_1 = 15°$) the selection process is nearly unaffected by the decoy nest-sites (which are seldom discovered at all) and the swarms will end up at either of the initially discovered nest-sites with the same likelihood, as one would expect in a situation where only the two discovered nest-sites are present.

As all the nest-sites in the experiment are of the same quality it does not matter for which site the swarm decides, as long as it decides. To test if close nest-site proximity obstructs a swarm's ability to choose the best available nest-site, we presented the swarm with 4 nest-sites. Again, these sites were located in opposing regions in pairs of two, with an angular separation of $\alpha_1 = 5°$ for sites in region 1 and $\alpha_2 = 15°$. Again, the swarm initially discovered one site in each region. The site discovered in region 2 was of good quality ($Q(S) = 75$), while all the other sites were of mediocre quality ($Q(s) = 50$).

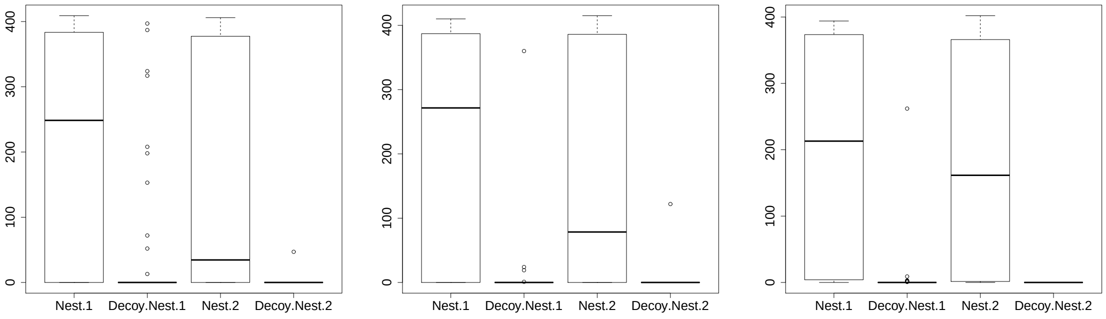
Figure 6.12 depicts the number of bees at each nest-site over the whole simulation run. In each of the 100 simulation runs the colony chose the site with the highest quality, while the other sites were abandoned. This indicates that spatial proximity does not hinder a swarm in choosing the best out of several options, in contrast to situations where multiple



(a) Separation $\alpha = 5°$
Low Quality ($Q(S) = 45$)

(b) Separation $\alpha = 5°$
Medium Quality ($Q(S) = 50$)

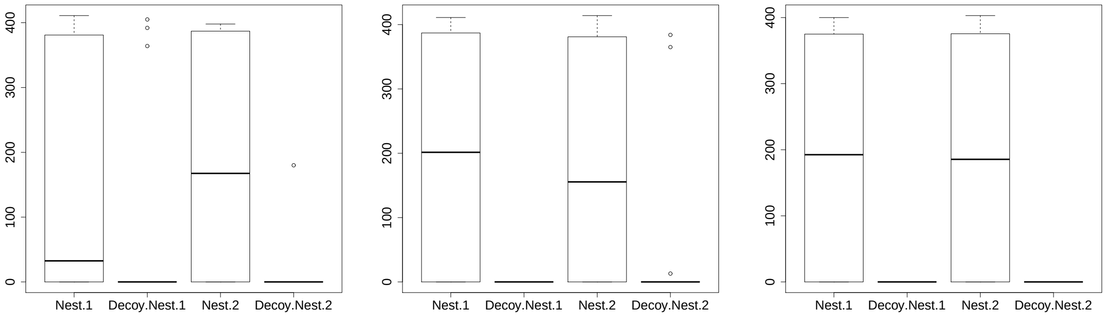
(c) Separation $\alpha = 5°$
High Quality ($Q(S) = 70$)



(d) Separation $\alpha = 8°$
Low Quality ($Q(S) = 45$)

(e) Separation $\alpha = 8°$
Medium Quality ($Q(S) = 50$)

(f) Separation $\alpha = 8°$
High Quality ($Q(S) = 70$)



(g) Separation $\alpha = 15°$
Low Quality ($Q(S) = 45$)

(h) Separation $\alpha = 15°$
Medium Quality ($Q(S) = 50$)

(i) Separation $\alpha = 15°$
High Quality ($Q(S) = 70$)

Figure 6.11.: Boxplot distribution of median number of bees on the 4 nest-sites after 57600 simulation steps (i.e., 16 hours).

options of same quality are in close proximity, which can delay a swarm's decision-making ability, as well as lead to a focusing of the selection process towards a particular nesting region.

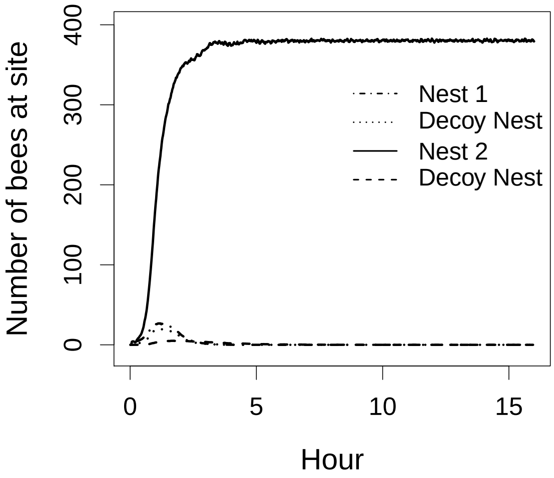


Figure 6.12.: Average number of bees assessing a nest-site over 57600 simulation steps (i.e., 16 hours). The swarm initially discovered two nest-sites which were located on opposite sites of the swarm: Nest 1 (mediocre quality) and Nest 2 (good quality). A decoy site was located next to each discovered nest-site, with differing angular separation: $\alpha_1 = 5°$ for Nest 1 and $\alpha_2 = 15°$ for Nest 2.

6.5. Conclusion

This chapter investigated the influence of spatial nest-site distribution on the nest-site selection process of the European honeybee *A. mellifera*. Our results suggest that the arrangement of nest-sites in the environment can indeed influence the selection process. Environments that exhibit a dense distribution of nest-sites can pose a problem for the selection process. This is due to the fact that bees that misread a waggle dance for a given site are likely to discover another site in the vicinity of the missed site, which leads to this site entering the selection process. As missing behaviour is unintended this can lead to a flux of recruits between nest-sites, thus preventing the swarm from converging towards choosing a single site. Two factors influence the impact of close-by nest-sites on the selection process.

The first factor is the angular separation between two sites regarding the swarm's location. Three angular separations $\alpha \in \{5°, 8°, 15°\}$ were tested here. If the angular separation between two sites is sufficient ($\alpha = 15°$), bees are very unlikely to discover additional nest-sites during their missing behaviour. With decreasing angular separation ($\alpha = 8°$) such sites are likely to be discovered and will enter the selection process, however as their discovery-rate is still low a swarm will be able to focus and converge towards a single nest-site. In situations of a very small angular separation ($\alpha = 5°$) missing behaviour does lead to a regular discovery of an additional nest-site thus obstructing the swarm's ability to converge towards a single site as the other site might always enter the selection process.

The second factor that influences a swarm's convergence of choice is the quality of the sites to be chosen. If the sites are of good quality this will lead to the bees promoting the sites performing more waggle dances on the swarm, which reduces the likelihood of followers missing the advertised site, and thus a site in close vicinity of the promoted site to enter the selection process.

When a swarm faces a decision between two nest-site regions with different nest-site densities the swarm will be biased towards the denser region if both regions contain nest-sites of equal quality. Again the swarm's preference depends on the angular separation between nest-sites. With increased separation the bias of the swarm towards a region will decrease until regional decoy nest-sites no longer influence the selection process and the swarm chooses uniformly between the nest-sites initially discovered. The ability of a swarm to choose the nest-site with the best quality is not affected by spatial setups and a swarm will always choose such a site even if initially presented with a nesting choice in a dense nest-site region.

These results suggest that a swarm that faces a decision between two regions of different nest-site densities will tend towards the denser nesting-region, thus the colony dispersion observed in the European honeybee seem to be a necessity imposed by the environment rather than a feature of the selection process. This may explains why high bee colony densities such as those reported by BAUM ET AL. (2005) and OLDROYD ET AL. (1995) occur in nest-site rich habitats.

As nest-site density influences the convergence speed of a swarm's decision making-process, our findings could also explain why open-nesting honeybee species such as *A. florea*, which faces nest-site selection in regions with abundant nesting locations, exhibit a fuzzy selection process (i.e., no quorum consensus is reached towards a site before lift-off), as this speeds up the selection process and prevents a swarm from getting stuck between decisions.

7. Swarm guidance in honeybees

In many animal species individuals exhibit collective movement, often over considerable distances, as they perform seasonal migrations, travel to food sources or return to safe havens (Boinski and Garber, 2000; Couzin and Krause, 2003; Krause and Ruxton, 2002; Simpson and Sword, 2010). The movement of these groups is commonly self-organized, arising from simple local interactions between individuals rather than from a command hierarchy. In many species, relatively few individuals within a group have pertinent information about the group's travel destination (usually because of differences among individuals in age or experience) and these informed individuals guide those that are not informed. The key to understanding group behaviour and its manifestations, such as crowd panic in humans (Helbing et al., 2007) or swarming in locusts (Bazazi et al., 2008), is to understand the nature of the local rules that individuals in the group follow, and to formalise these in simulation models (Sumpter, 2010).

In the case of honeybees the question of group guidance has been studied experimentally, in particular in one species, the European honeybee *A. mellifera*. Group guidance follows the successful selection of a new nest-site by a reproductive swarm (please refer to chapter Chapter 5 for a biological outline of nest-site selection). Once the swarm has achieved a quorum for a given nest-site, bees that were involved in the quorum return to the swarm and prepare its lift-off using an auditory signal known as piping. This signal informs the quiescent bees in the cluster that they should prepare themselves for flight (Seeley et al., 2003). The final signals for flight are "buzz running", in which a scout runs in zig-zags over the swarm vibrating its wings every second or so (Lindauer, 1955). The swarm then takes flight and flies to its chosen home guided by the bees that know the location of the new nest.

How are honeybee swarms guided? Two mechanisms have been proposed. Lindauer (1955) observed in airborne swarms that some bees fly through the swarm cloud at high speed and in the correct travel direction, seemingly pointing in the direction of the new nest-site. He suggested that these fast-flying bees, "streakers", are scouts that have visited the chosen nest-site and that their behaviour guides the other uninformed bees towards their new home. This hypothesis has been referred to as the streaker or vision hypothesis. An alternative is the olfaction hypothesis of Avitabile et al. (1975), who proposed that the scouts provide guidance by releasing assembly pheromone from their Nasanov glands (a gland on the bees abdomen) on one side of the swarm cloud, thereby creating an odour gradient that guides the other bees in the swarm. BEEKMAN ET AL. (2006) tested both the vision and the olfaction hypotheses in honeybee swarms. They studied in detail the flights of normal honeybee swarms and swarms in which each bee's Nasanov gland was sealed shut. Their results firmly reject the olfaction hypothesis, as the sealed swarms were perfectly able to fly to their destination. Although this study does not provide direct proof for the vision hypothesis, it obtained evidence strongly consistent with it, as it found that the peak flight speeds of swarms (2-3 m/s) are well below the peak flight speeds of individual bees (9-10 m/s). This shows that it is possible for scout bees to streak through a flying swarm. Additionally, using photographic analysis BEEKMAN ET AL. (2006) were able to show that a moving bee swarm contains fast-flying individuals in its upper half.

Since BEEKMAN ET AL.'s first experimental study on the guidance of honeybee swarms, two more experimental studies have investigated been performed (LATTY ET AL., 2009; SCHULTZ ET AL., 2008). SCHULTZ ET AL. (2008) provided further information about the speed and directionality of bees in a moving swarm. In agreement with BEEKMAN ET AL. (2006), their analysis revealed that the swarm contains fast flying individuals heading towards the desired nest-site. Additionally they observed that the distribution of individual speed and flight angles differs within the swarm. While individuals in the upper half in the swarm seem more aligned towards the swarm's travel goal, individuals in the bottom of the swarm are not well aligned in terms of directionality. This study indicates that a moving bee swarm is far from directional consensus on an individual level, while still being able to move towards a given location as a whole. LATTY ET AL. (2009) investigated the impact of directional noise, provided by fast flying foragers, on a swarm's movement. By setting up a "bee highway" (i.e., a foraging route that ran orthogonal to the swarm's travel route), they were able to show, that an airborne bee swarm is significantly handicapped by fast foragers that travel orthogonally to the swarm's heading direction.

The experimental evidence outlined above clearly favours streaking as the mechanism underlying honeybee migration. Another possible explanatory guidance principle in honeybee swarms that has not yet been ruled out is passive guidance (COUZIN ET AL., 2005). In such a situation the group is guided by a few informed individuals without these individuals providing explicit guidance signals that would allow a group-intern distinction between informed and uninformed individuals.

In order to decide which of these competing mechanisms (passive / active guidance) best explains honeybee swarming behaviour, evidence must be shown that the candidate mechanisms can reproduce features that are observed in nature in moving swarms. As pointed out in SCHULTZ ET AL. (2008) bee swarm is far from being an aligned entity, in contrast to fish shoals. This chapter thus investigates to what extent active and passive guiding mechanisms can account for directional noise within a moving group and thus reproduce known features of a migrating honeybee swarm.

A second aspect we will investigate in this Chapter is to what extent the streaking hypothesis requires directional consensus in order to lead to directed group motion. As pointed out in Chapter 5 differences in nesting requirements between open and cavity nesting honeybee species impact the frequency with which they will search for a new home, as well as the accuracy of the decision-making process. While reproductive swarms of the European honeybee *A. mellifera* will only lift off after a consensus for a single site has been achieved, swarms of open-nesting species such as *A. florea* will start moving before that is the case.

Given that *A. florea* swarms lift-off without having reached a consensus on the precise direction in which to fly (MAKINSON ET AL., 2011), a universal flight mechanism underlying honeybee migration would require the ability to lead to a directed flight even under directional dissent. In the case of passive guidance (COUZIN ET AL., 2005), it has already been shown that different directional preferences do not hinder swarm guidance. For the streaking hypothesis this has not been investigated yet. If the streaking hypothesis proofs being able to cope with directional dissent this would be an indication that it can serve as a fundamental movement hypothesis for the whole Apis genus and not only for specific species.

7.1. Group guidance in animal collectives

One of the first models of collective movement proposed by REYNOLDS (1987) demonstrated that the coordinated movement of a collective can be achieved by each individual aligning its position and direction of movement with that of individuals within a certain neighbourhood, while keeping a minimum distance to its immediate neighbours. In a model this can be achieved by a set of simple uniform rules representing attraction to neighbours with respect to vectorial alignment and avoidance. Such an allelomimetic mechanism ("do what your neighbour does") has been shown, both theoretically and empirically, to lead to collective movement and explain patterns observed in nature (SUMPTER, 2010). Given the diversity of animal groups, it is not surprising that a range of models has been developed to study collective movement of animal groups (CRESSWELL ET AL., 2011; VICSEK AND ZAFIRIS, 2010). These models share the core assumptions of allelomimetics while the exact implementation of the movement rules differ depending on the species studied.

There are two extreme ways in which groups can "decide" on a direction of movement. Either all individuals within the group contribute to a consensus, or else relatively few individuals (for convenience we will call these leaders) have information about the group's travel destination and guide the uninformed majority. In some species all individuals within a group share a genetically determined propensity to travel in a certain direction (Berthold et al., 1992; Berthold and Querner, 1981) or all are involved in choosing a particular travel direction (Grünbaum, 1998; Neill, 1979). In other species groups are guided by a small number of individuals that determine the group's foraging movements and steer a group towards a target (Reebs, 2000; Seeley et al., 1979; Swaney et al., 2001).

For groups containing only a small fraction of leaders, the question arises, how information on travel direction is dispersed throughout the swarm in order to produce a directed movement of the group. Recently, two theoretical studies have addressed the issue of information transfer from informed to uninformed group members. Janson et al. (2005) modelled a situation of active guidance, where informed individuals make their presence known by moving at a higher speed than the average group member and in the direction of travel. Uninformed individuals tend to align their direction of movement with that of their neighbours, taking individuals moving faster than themselves strongly into account. As informed individuals initially move faster, they have a larger influence on the directional movement of the uninformed individuals, thereby steering the group and allowing guidance to emerge.

Couzin et al. (2005) have shown that group guidance can also be achieved passively. In such a situation the group is guided by a few informed individuals without these individuals providing explicit guidance signals that would allow a group-intern distinction between informed and uninformed individuals. Informed individuals differ from uninformed ones only in the tendency to move in a preferred direction. Given this directional preference and the tendency of the group to stay together, informed individuals will passively steer the group towards their directional preference.

The main difference between the two models lies in the presence or absence of cues or signals from the informed individuals to the uninformed majority. Janson et al.'s leaders clearly make their presence known, whereas Couzin et al.'s model suggests that leadership can arise simply as a function of information difference between informed and uninformed individuals, without the individuals being able to tell which ones have more information.

7.2. Swarm guidance model

Here an extended and revised version of the streaking model proposed by Janson et al. (2005) was used to investigate how well the two different paradigms (i.e., passive and active guidance) resemble the dynamics empirically observed in moving bee swarms. Furthermore we tested to what extent the streaking mechanism is able to cope with directional dissent among streakers.

As outlined above, recent experimental studies have investigated the individual dynamics underlying a flying bee swarm. SCHULTZ ET AL.'s detailed photographic analysis of a moving bee-swarm provided information about the distribution of individual flight-speed and directionality in a moving swarm (SCHULTZ ET AL., 2008). JANSON ET AL.'s original model lacks these dynamics – although the swarm is guided by streakers, individuals in a moving swarm are too well-aligned for a moving bee swarm. Additionally, the resulting streaker model is further modified to model swarm movement under passive guidance. We modify, rather than create a new or use an existing model in this study in order to compare the two approaches with the same set of parameters. In the following the extended and revised version of the streaker model will be outlined.

At any time point t in the simulation each individual i of the swarm is represented by two 3-dimensional vectors, one denoting its current position $p_i(t)$ and the other one its velocity $v_i(t)$. In the following formal description of the model the index i will be omitted providing that the context is clear. In accordance with JANSON ET AL. (2005) informed individuals are guided by different rule-sets in the case of active guidance (streaking), whereas all bees show more homogeneous behaviour in the case of passive guidance (COUZIN ET AL., 2005).

7.2.1. Active guidance: Behaviour of uninformed individuals

The idea behind the guided flight behaviour is that uninformed bees are influenced by movements in their surrounding. An uninformed individual's behaviour is guided by four rules:

- *Avoid*: The avoidance rule prevents collisions between individual's in the swarm and reflects an individuals intention to keep a certain distance between itself and other individuals.
- *Align*: The alignment rule enables individuals to adjust their orientation to the heading direction of neighbouring individuals and thus enables a guided movement of the swarm.
- *Cohere*: The coherence rule denotes the tendency of individuals to stick together and thus prevents the swarm from breaking up.
- *Random*: The random rule adds noise to an individual's movement.

Each rule results in a vector whose weighted sum corresponds to an individual's velocity update, with the weights reflecting the influence of the respective rule. In the following the rules are outlined more formally.

Avoid The avoidance vector v_{avoid} implements the tendency of an individual to move away from neighbours that come too close. It is calculated in the same manner as in the model of JANSON ET AL. (2005):

$$v' = \frac{1}{d_{min}} \cdot \frac{1}{|N_{min}|} \cdot \sum_{j \in N_{min}} (p - p_j) \cdot (\frac{d_{min}}{p - p_j} - 1) \tag{7.1}$$

$$v_{avoid} = \frac{v'}{|v'|^{\alpha}} \tag{7.2}$$

N_{min} denotes the set of neighbours the focal individual tries to avoid, because they violate its personal space (defined by a minimum distance d_{min}). Similar to force fields, the magnitude of the avoidance correlates to the closeness of an individual (i.e., the closer the stronger the avoidance, see fourth factor in Eq. 7.1). By scaling v' via Eq. 7.2 with $\alpha \in (0, 1)$ the avoidance vector is further increased, however its length will remain within $[0, 1]$.

Align As pointed out above, the alignment rule enables a swarm to direct its movement in a specific direction. The alignment rule used here differs from the one used by JANSON ET AL. (2005). We use a topological neighbourhood metric (i.e., the k closest individuals) instead of a Euclidean metric (i.e., all individuals within a certain range). Topological metrics have been used to model the flocking behaviour of sparrows (e.g., BALLERINI ET AL. 2008). These metrics have the benefit that the size of an individual's neighbourhood stays stable regardless of the distance between it and its neighbours. This prevents over-information as well as loosing track of neighbours. Additionally, only faster flying individuals are incorporated in the alignment, while all neighbours were taken into account previously (JANSON ET AL., 2005).

The alignment vector is calculated as follows: First, the k closest neighbours of a given individual i are determined. This results in a set of neighbours K_i. Let $s_i = |v_i|$ denote the speed of an individual i, then $K_i^{fast} = \{k_j \mid k_j \in K_i \text{ and } s_j \geq 2 \cdot s_i\}$ constitutes the set of neighbours that are at least twice as fast as individual i. For its alignment, an individual will only take K_i^{fast} into account (no alignment will take place if $K_i^{fast} = \emptyset$). The alignment vector is calculated as

$$v_{align} = \frac{1}{v_{min}} \cdot \frac{1}{|K_i^{fast}|} \cdot \sum_{j \in K_i^{fast}} v_j \tag{7.3}$$

The first factor of Eq. 7.3, scales the length of v_{align} between [0,1] if the speed of the individuals in K_i^{fast} is lower than a given minimum speed v_{min}. Such a scaling is necessary to prevent the perpetuation of alignment (and thus movement of the swarm) in a system where no informed individuals (i.e., streakers) are present. Once streakers are active or neighbours reach a speed greater than v_{min}, they will have a stronger impact on the alignment vector v_{align}.

Cohere The centre of a swarm containing n individuals is defined as

$$p_{core} = \frac{1}{n} \cdot \sum_{j=1}^{n} p_j \tag{7.4}$$

the coherence vector of a given individual i is thus given by

$$v_{cohere} = \begin{cases} \frac{1}{d_{core}} \cdot (p_{core} - p_i) & \text{if } K_i^{fast} \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \tag{7.5}$$

d_{core} is a scaling factor that limits the length of v_{cohere} to $[0,1]$ if an individual is within d_{core} distance to the swarm's centre. The closer an individual is to the swarm's centre, the less need it feels to get closer. Thus, individuals at the outside of the swarm will cohere more strongly than those in the centre.

In contrast to the previous model (JANSON ET AL., 2005), coherence is only applied if faster flying individuals are present in an individual's neighbourhood. This change was necessary to fit the model to experimental data (SCHULTZ ET AL., 2008), which showed that individuals in a swarm behave more independently (in terms of flight speed and angle) than initially expected. If coherence and alignment would be applied regardless of speed (even when strongly weighting streakers) the streakers' direction would propagate through the swarm, leading to a very strong directional alignment within the whole swarm. In order to maintain a level of individuality within the swarm, individuals here will only be governed by alignment and coherence if they are disturbed by significantly faster flying individuals or leave the swarm. When an individual leaves the swarm it will slow down its movement, making it more receptive to alignment and coherence which will lead it back to the swarm and prevent a swarm breakup.

Random The random vector introduces randomness into the velocity of an individual and is achieved in the same manner as in JANSON ET AL. (2005)

$$v_{random} = \beta \cdot \frac{v''}{|v''|} \tag{7.6}$$

v'' is randomly chosen from $[-1,1]^3$ and the scaling factor β is chosen randomly according to the distribution function $F(x)$ of the exponential distribution $F(x) = 1 - e^{-\lambda \cdot x}$ restricted to $[0,1]$ with $\lambda = 2$.

Update For the velocity update the weighed sum of these vectors is calculated

$$v^*_{new} = w_{cohere} \cdot v_{cohere} + w_{avoid} \cdot v_{avoid} + w_{align} \cdot v_{align} + w_{random} \cdot v_{random} \tag{7.7}$$

where w_{cohere}, w_{avoid}, w_{avoid}, w_{align} and w_{random} denote positive weights of the respective vectors. The weights enable us to emphasize certain factors, which is done here in contrast to the previous model (Janson et al., 2005) where each rule had the same impact on an individual's behaviour (for specific parameter setting please refer to Section 7.3). The length of v^*_{new} constitutes an individual's change in velocity. In order to prevent infinite speed-up the acceleration of an individual per simulation step is capped using a maximum acceleration a_{max}

$$v_{new} = \begin{cases} v^*_{new} & \text{if } |v^*_{new}| < a_{max} \\ \frac{v^*_{new}}{|v^*_{new}|} \cdot a_{max} & \text{otherwise} \end{cases} \tag{7.8}$$

In order to calculate the new velocity of an individual a fraction $w \in [0, 1]$ of the old velocity is kept and the update velocity is added

$$v(t+1) = w \cdot v(t) + v_{new} \tag{7.9}$$

After the velocity is updated the position of each individual is derived according to $p(t+1) = p(t) + v(t+1)$.

7.2.2. Active guidance: Behaviour of informed individuals

Informed bees (streakers) guide a swarm by consecutive fast flights through the swarm. Here they streak through the swarm along a straight flight path, which is parallel to the line that goes through the swarm centre in their preferred direction of travel. Once streakers reach the front of the swarm (i.e., they have less than 10 surrounding neighbours within a given distance $d_{vis} = d_{min}$), they will fly back to the trailing edge of the swarm. Reaching the end of the swarm (i.e., having less than 10 surrounding neighbours in a given distance d_{vis}) will trigger their streaking again.

Empirical data suggests that streaking occurs in the upper segment of a swarm while bees appear to fly back through the lower section of the swarm (Schultz et al., 2008). We thus implemented streaking and return flights at different heights within the swarm. Thus streakers that reach the front of the swarm are relocated to the bottom front of the swarm where they will start to fly back. Accordingly streakers that reach the back of the swarm will be relocated to the upper segment of the swarm before streaking is re-initiated. Streakers will use different speeds for streaking (s_{streak}) and flying back ($s_{flyback}$).

7.2.3. Passive guidance: Behaviour of uninformed individuals

One argument against the streaker hypothesis is that it requires two distinct behavioural patterns, while group guidance in animal collectives can also be achieved in a more homogeneous, thus simpler manner.

While speed differences play a crucial role under the streaker hypothesis (as faster-flying neighbours have a strong impact on an individual's orientation), this is not the case for models of passive guidance. All individuals in a swarm exhibit the same constant speed and potential neighbours are not discriminated according to speed differences.

In order to transform the streaker model outlined above into a model of passive guidance, several changes have to be made. Passive guidance can be easily achieved by modifying the *align*, *cohere* and *update* rules of the streaker model.

Align The alignment rule has to be changed as it now has to disregard the speed of neighbouring individuals. Again, the k individuals that are closest to an individual are considered to be its neighbourhood. The alignment vector is then calculated as

$$v_{align} = \frac{1}{|K_i|} \sum_{j \in K_i} \frac{v_j}{|v_j|} \tag{7.10}$$

In order to make sure that each neighbour has the same influence on an individual, the unit vectors of the velocities rather than the velocities themselves are used.

Cohere The coherence rule has to be changed as well. While coherence is only executed if fast neighbours are present, it will apply each time step in a model of passive guidance. This leads to coherence having a too strong impact on the model when it is used in streaker form. The reformulation of v_{cohere} will return a unit vector and this reduces the impact on the velocity update. this is in contrast to Eq. 7.5, where the length of the vector can exceed 1, depending on the individual's distance to the swarm core.

$$v_{cohere} = \frac{(p_{core} - p)}{|(p_{core} - p)|} \tag{7.11}$$

Update In contrast to the streaker approach, all individuals exhibit the same constant speed under passive guidance. Thus after calculating $v(t+1)$ (see Eq. 7.7 and 7.8) the new velocities need to be normalized regarding a desired speed s.

$$v(t+1) = \frac{v(t+1)}{|v(t+1))|} \cdot s \tag{7.12}$$

Finally the position of each individual is updated according to $p(t+1) = p(t) + v(t+1)$

7.2.4. Passive guidance: Behaviour of informed individuals

While streakers exhibit a different behavioural repertoire than uninformed swarm members, this is not the case under passive guidance. Here the guides only differ by incorporating yet another vector that describes their tendency to go towards a desired destination p_{dest}

$$v_{dir} = \frac{p_{dest} - p_i}{|p_{dest} - p_i|} \tag{7.13}$$

which is then incorporated into the velocity sum update with a given weight w^i_{dir}

$$\begin{aligned} v''_{new} = w_{cohere} \cdot v_{cohere} + w_{avoid} \cdot v_{avoid} + \\ w_{align} \cdot v_{align} + w_{random} \cdot v_{random} + w_{dir} \cdot v_{dir} \end{aligned} \tag{7.14}$$

The component velocities of informed individuals are updated in the same manner as for uninformed individuals according to Eq. 7.7, 7.8 and 7.12.

7.3. Experimental setup

Following Janson et al. (2005), each swarm was initialized by placing its individuals at a random position within a cube of side length $n/10$ cm, where n corresponds to the number of uninformed bees in the swarm and the cube is centred around a starting position (200,200,200) (note that one unit corresponds to 1 centimetre). Each simulation step corresponded to 10 milliseconds of realtime. Such a fine scale was necessary in order to capture the dynamics of the system. A simulation run lasted for 2000 simulation steps (i.e., 20 seconds) with the first 300 simulation steps corresponding to an initialization phase. During this initialisation phase each bee, streakers and uninformed bees alike, showed the same behaviour. To allow for a realistic bee distribution in the swarm, only the *avoid*, *cohere* and *random* rules were applied for the first 150 steps of the initialization phase when updating each individual's position and velocity. The alignment rule was enabled in the second half of the initialization phase. Guidance started after the initialization.

Table 7.1 lists the parameter values used in the experiments comparing active and passive guidance. As can be seen, all parameters are identical except for the weight of the random rule w_{random} are identical.

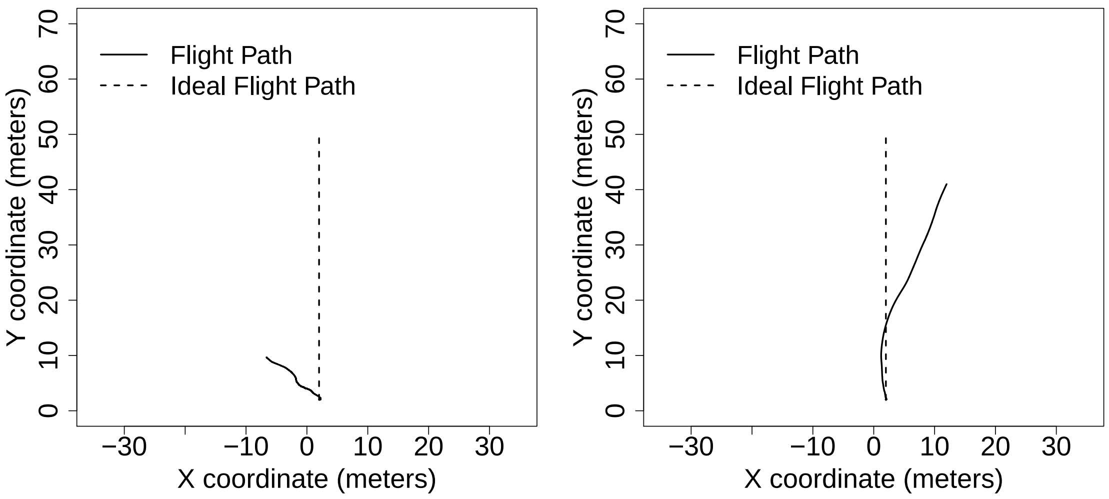
The weight of the random rule w_{random} had to be lowered in order to achieve a directed flight behaviour during passive guidance. Figure 7.1 depicts the representative flight trajectories of a passively guided swarm under both random weight settings $w_{\text{random}} \in \{0.2, 0.6\}$ over a whole simulation run. Informed individuals would like to reach the point (200,10000,200) and the ideal flight path thus corresponds to a vertical trajectory along the y-axis.

Table 7.1.: List of parameter descriptions and values used in the streaker model. Value (Active) denotes the parameters used for active guidance. Value (Passive) denotes the parameters used for passive guidance.

Parameter	Description	Value (Active)	Value (Passive)
n	number of uninformed bees	475	475
n_{inf}	number of informed bees	25	25
w_{avoid}	weight of avoidance vector	1.0	1.0
w_{align}	weight of alignment vector	0.3	0.3
w_{cohere}	weight of coherence vector	0.3	0.3
w_{random}	weight of random vector	0.6	0.2
k	number of nearest neighbours for alignment	10	10
α	exponent used in scaling of avoidance vector	3/4	3/4
v_{min}	minimum speed required for alignment	2.0	–
a_{max}	maximum individual acceleration	20 cm/10ms	–
d_{min}	visibility length scale	20 cm	20 cm
d_{core}	scaling factor for distance to swarm core	20 cm	20 cm
a_{informed}	speed of informed bees	9.55 cm/10ms	–
a_{flyback}	flyback speed of informed bees	3.55 cm/10ms	–
w_{dir}	preferred direction preferred of informed bees	–	0.3
s	speed of individuals	–	2.7 cm/10ms

Regardless of the strength of the noise weight, the swarm is able to stick together. However, as can be seen in Figure 7.1(a), passive guides are not able to set a swarm in the desired direction given too much individual noise in the swarm (i.e., $w_{\text{random}} = 0.6$), while the this is possible given the smaller random noise weight (i.e., $w_{\text{random}} = 0.2$). The reason for this lies in the fact that under passive guidance an individual can not discriminate between individuals and will thus take all of it's neighbours into account. This leads to the case that an individual's neighbours will influence its directional update in each step (i.e., regardless of their speed). As the random weight essentially puts noise in an individual's directionality, this leads to a decrease in the broadcast of correct directional information throughout the swarm, as such information will become increasingly distorted over consecutive simulation steps. The swarm will thus not be able to travel in the direction desired by the informed individuals. This is also reflected in the distance a swarm is able to move under the differing random weight settings. While the swarm is able to travel around 11.5 meters given a random weight of $w_{\text{random}} = 0.6$, it travels approximately 40 meters with the lower random weight $w_{\text{random}} = 0.2$.

For the experiments investigating the necessity of directional dissent under the streaking paradigm a smaller number of informed individuals was used than for the experiments



(a) $w_{\text{random}} = 0.6$

(b) $w_{\text{random}} = 0.2$

Figure 7.1.: Sample flight trajectory of passively guided swarm under differing random weight settings $w_{\text{random}} \in \{0.2, 0.6\}$.

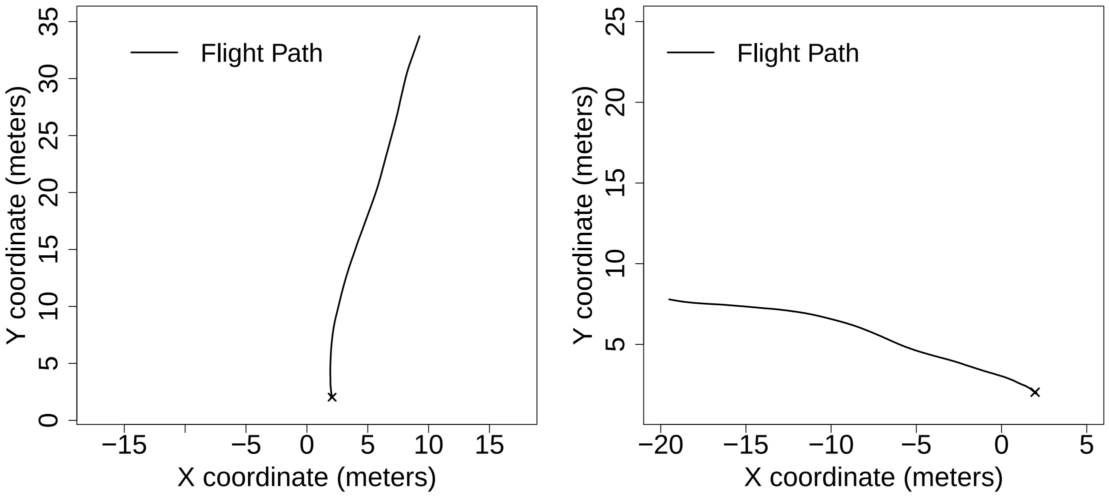
comparing active and passive guidance. Based on Makinson et al. (2011)'s *A. florea* data, it was assumed that only 2% of the individuals would engage in the streaking process. This leads to $n_{inf} = 10$ informed individuals and $n = 490$ uninformed individuals given a total swarm size of 500. A swarm size of 500 is however relatively small in comparison to the size of real *A. florea* swarms. In order to demonstrate that the streaking hypothesis is also applicable to swarms with real *A. florea* characteristics, the model was used in a final experiment to simulate the flight of test swarm 1 of Makinson et al. (2011). The exact size of swarm 1 was not estimated in the original study, but it appeared to be roughly the same size as swarm 5 (Makinson et al., 2011). Accordingly, for the simulation of a real *A. florea* swarm-flight, a swarm size of 2700 individuals was used. Here it is assumed that the individuals that danced in the last hour before lift-off (i.e., 28 individuals) will act as streakers during the swarm's flight. Each experiment was repeated 5 times.

7.4. Experiments: Active vs passive guidance

As pointed out earlier the study by Schultz et al. (2008) revealed that a moving swarm of honeybees is quite noisy in terms of individual directionality and speed. The following experiments test to what extent passive and active guidance can achieve such characteristics. In particular we will compare the flight-behaviour of a swarm containing informed individuals with one that does not contain informed individuals. Additionally, the angular alignment within a swarm under the two moving hypothesis will be compared with SCHULTZ ET AL.'s findings. The preferred direction of the informed individuals within these experiments was 0°, which corresponds to a vertical flightpath along the y-axis, starting from the initialization point of the swarm.

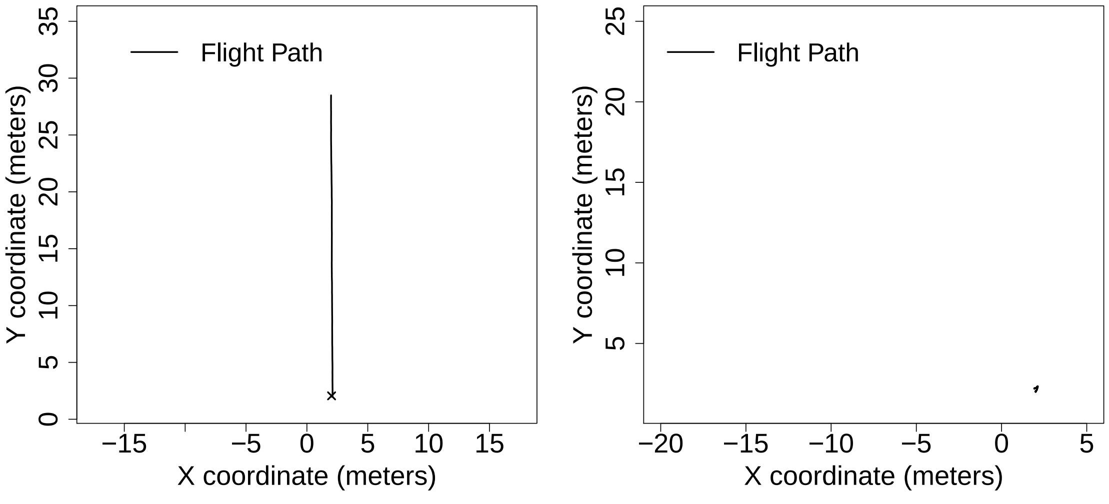
7.4.1. Swarm behaviour with and without informed individuals

It is assumed that a honeybee swarm will only set flight when informed individuals are present in the swarm and a decision has been made. A honeybee swarm that does not contain any leaders will not be able to find any suitable nest-site and should thus recluster and reinitiate the decision-making process rather than perform an aimless flight. Unfortunately the flight-behaviour of a leaderless honeybee swarm has not been tested yet experimentally, as it is simply impossible to remove informed individuals from an airborne swarm. As informed individuals are those who prepare a swarm for lift-off it is also not possible to get a swarm airborne when no informed individuals are present. A situation similar to this is when the airborne swarm does not contain a queen and it does not make sense to fly to a new nest-site as the core of the new colony is absent. In such a case the swarm will not travel in any direction. Instead it will aimlessly hover around its previous cluster location and settle again (personal communication Madeleine Beekman). To find out what impact the two different movement strategies have on a swarm's flight trajectory with and without informed individuals we conducted two flight experiments for each guiding strategy, one with informed individuals and one without informed individuals. Figure 7.2 depicts the average flight trajectory of a passively guided swarm over 5 repeats. As can be seen the swarm will always set in motion, regardless of whether informed individuals are present or not. However, the flight direction depends on the presence of informed individuals. The reason why a swarm that does not contain any informed individuals will still start to move is that the swarm does not require faster flying individuals to speed up. As individuals have no means of discriminating between informed and uninformed individuals in a passively guided swarm, they will cohere and align to their neighbours. The swarm will thus reach a random consensus direction at some point in which it will fly. Adding informed individuals (Figure 7.2(a)) only adds an directional bias which will pull the swarm towards the target direction and prevents it from flying in a random direction. As can be seen, the influence of the uninformed individuals on the swarm's flight trajectory is still present when informed individuals guide the swarm, as they not only influence other uninformed individuals but also the informed ones. Active guidance differs in this respect. As can be seen in Figure 7.3 a swarm will only take flight, if informed individuals are present in the swarm. This is due to the fact that informed individuals not



(a) Informed individuals present (b) No informed individuals present

Figure 7.2.: Average flightpath over 5 simulation runs of a passively guided swarm with (a) and without (b) informed individuals. Crosses indicate the position of the swarm at the end of simulation step 150 after alignment was enabled.



(a) Informed individuals present (b) No informed individuals present

Figure 7.3.: Average flightpath over 5 simulation runs of an actively guided swarm with (a) and without (b) informed individuals. Crosses indicate the position of the swarm at the end of simulation step 150 after alignment was enabled (in Figure (b) the cross was omitted to expose the flight-path).

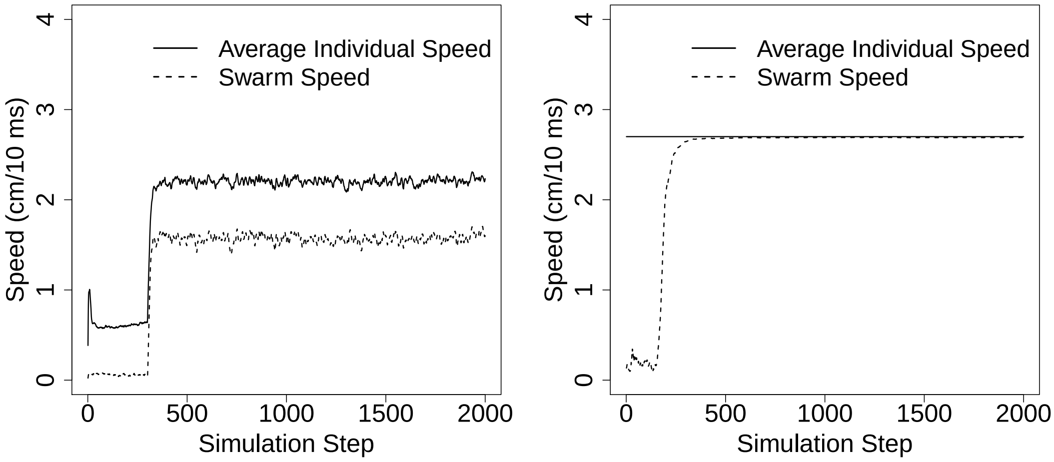
only set a swarm's directionality but also initiate its speed-up. Uninformed individuals in the presented active guidance model are very wayward, which means that when no faster individual is among their 10 closest neighbours they will only try to stick to the swarm via the coherence rule and avoid other individuals. In order to get the swarm moving it thus requires informed individuals as these will initiate directed and fast movement in the swarm. The impact of active guides is also notable when comparing the flight trajectories of passively and actively guided swarms. Uninformed individuals clearly influence the flight trajectory in the case of passive guidance, which leads to a swarm movement in the preferred direction albeit not close to the ideal flight trajectory. In contrast, the trajectory of an actively guided swarm is quite close to the optimal flight trajectory even though uninformed individuals exhibit a higher noise level and are thus more reluctant to fly on route than uninformed individuals in a passively guided swarm.

7.4.2. Flight speeds of a guided swarm

While analysing the flight characteristics of a moving bee swarm, SCHULTZ ET AL. (2008) noticed that the flight speed of individual bees within the swarm differed depending on the bee's position and orientation. Additionally, they found that individual bees move significantly faster than the complete swarm, which is an indicator for poor directional alignment in the swarm, as under complete alignment the swarm's velocity should correspond to the velocity of each individual.

Figure 7.4 depicts the evolution of swarm speed and average individual speed under the two different guiding mechanisms. When comparing the characteristics of the first 500 simulation steps the differences between the two guiding mechanisms are quite obvious. Swarm and individual movement under active guidance clearly depends on the presence of informed individuals. The activation of the alignment rule at simulation step 150 has no effect on the swarm's or the individuals' movement speed (the initial peak in Figure 7.4(a) is due to the fact that individuals are initialized quite close to each other and thus first need to establish some personal space). Only when the streakers are added to the swarm (i.e., at simulation step 300), both the individuals' and the swarm's speed increase and the swarm sets in motion. As in the empirical data of SCHULTZ ET AL. (2008), the average individual flight speed exceeds the swarm's movement speed.

Under passive guidance the alignment rule governs the flight behaviour of a swarm. A swarm will set in motion as soon as the alignment rule is activated (i.e., at simulation step 150), which explains the flight trajectories found for an unguided swarm, which were presented in the last section. In contrast to active guidance the swarm will reach the same speed as the fixed individual speed of (2.7 cm/10ms).



(a) Active guidance

(b) Passive guidance

Figure 7.4.: Average individual and swarm velocity development in the time course of a simulation under active (a) and passive (b) guidance.

From a speed perspective, active guidance is thus favourable in terms of resembling characteristics of a moving honeybee swarm.

7.4.3. Angular distribution within a guided swarm

Schultz et al. (2008) also estimated the flight-angle distributions for different parts of a flying swarm. The swarm was partitioned into bottom and top. In each of these sections the flight angles in the front, middle and back were measured, resulting in a total of 6 flight angle distributions. To gain the data a moving swarm was repeatedly photographed while flying over a camera (Schultz et al., 2008). Their findings show that the variance of the flight angle distributions in the top of the swarm is smaller within all sections than the flight angle distribution in the bottom of the swarm. Especially the bottom middle of the swarm exhibited a wide range of individual orientation, meaning that the bees were not well aligned to each other at all in this segment. Here we test to what extent such angular distributions can be achieved in simulation under different guidance strategies.

To investigate the angular distribution within a swarm, individuals' orientations at a single timepoint were used. In order to ensure that the observed angular distributions at a single timestep are meaningful for the whole flight trajectory of the swarm, a Phillips-Perron Test for Unit Roots (Perron, 1988) was conducted on the angular deviations from the swarm's mean direction between timestep 600 and 2000. Timestep 600 was chosen as a starting point of the time-series as it is assumed that the swarm reached a stationary state at that timepoint after ending its initialization phase at timestep 300. The Phillips-Perron tests confirmed that the angular deviation time-series of our simulations do not contain a unit root, which allows the assumption that the flight angles between timesteps 600 and 2000 are stationary and thus that observations made at one timestep can be generalized regarding the whole simulation after timestep 600.

As in SCHULTZ ET AL. (2008) the individual flight angles were grouped into six categories (top/back, top/middle, top/front, bottom/back, bottom/middle, bottom/front), depending on an individual's position in the swarm. An individual was considered to be in the top of the swarm if it was above the swarm core. To establish the back, middle and front regions of the swarm a rectangle was fitted around the swarm in the orientation of the informed individuals' preferred direction. The sides parallel to the optimal flight path were divided into 3 equal sections corresponding to back, middle and front respectively.

Figure 7.5 depicts the individuals' orientations in a swarm under different guiding strategies. The angular distribution between the two different guiding strategies differs significantly (circular Kruskal-Wallis test, $\alpha = 0.05$[1]). As can be seen when comparing Figure 7.5(a) with 7.5(b), an actively guided swarm shows stronger directional dissent than a swarm guided passively, which corresponds to and thus better resembles the dynamics that SCHULTZ ET AL. observed in their experiment. As in the experimental data, directional dissent is strongest in the middle bottom segment of an actively guided swarm, which is not the case for a passively guided swarm.

Another finding of SCHULTZ ET AL. (2008) is that the angular variance differs between the top and the bottom segment of the swarm. Table 7.2 exhibits the variances of angular orientation (in radians) in the six different segments of the swarm under both active and passive guidance. As can be seen, both guiding mechanisms exhibit a difference in angular variance, with the difference between top and bottom of the swarm being of greater magnitude for actively guided swarms. To test if the observed difference in angular variance between the top and bottom segment of a swarm is statistically significant a circular Kruskal-Wallis test ($\alpha = 0.05$) was performed. The difference between the angular distributions in the top and bottom segments of the swarm is significant for both guiding mechanisms. However as the flight angle variance observed in the empirical data is quite large between the top and bottom segment of the swarm, the flight angle characteristics of a moving honeybee swarm are better characterized by an actively guided swarm.

It should be pointed out that the directional dissent measured by SCHULTZ ET AL. (2008) is of higher magnitude than that observed in simulations of either guiding mechanism, which suggests that real bees are even more wayward than our virtual model bees. The reason why active guidance is able to reflect a real swarm's characteristics better than

---

[1] All statistical evaluations on the angular data were performed with the circstat MATLAB toolbox (BERENS, 2009)

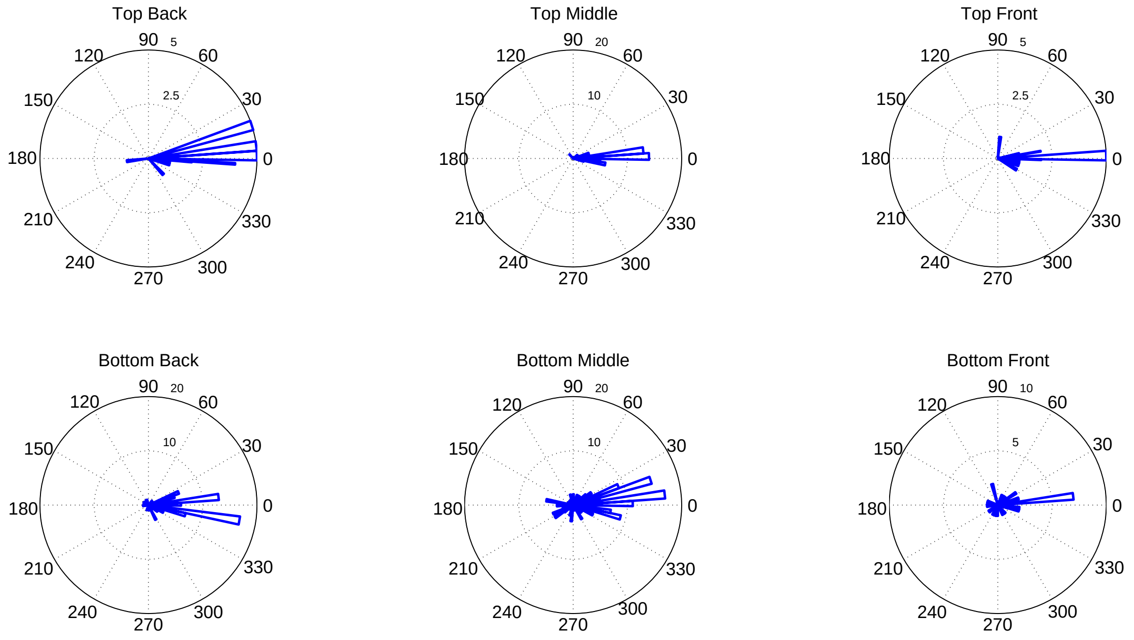
passive guidance is due to the nature of these two guiding strategies. As outlined earlier, guidance is usually thought to be based on allelomimetics (i.e., do what your neighbour does), which is bound to create alignment within a moving group. As an individual is not able to distinguish between informant and and non-informant individuals under passive guidance it will take everyone into account, which will increase the allelomimetics effect and thus create strong alignment. While such guiding strategies resemble the travel trajectory of several species perfectly well (SUMPTER, 2010), this seems not to be the case for honeybees. The angular noise within a colony is a strong indicator that honeybees are very reluctant to do as their neighbours do and need guides that catch their attention (i.e., by streaking) in order to get the swarm moving in the right direction. This has the advantage that an uninformed bee's personal flight preference will not propagate through the swarm, and a swarm is thus able to fly towards a location more precisely. As nest-sites are relatively small in comparison to the distance a swarm has to travel towards a new site, precision is crucial, as only a small angular deviation within a swarm's flight trajectory might lead to the swarm ending up at an unwanted location.

Table 7.2.: Variances of angular orientation (in radians) within different segments of a swarm under differing guidance strategies.

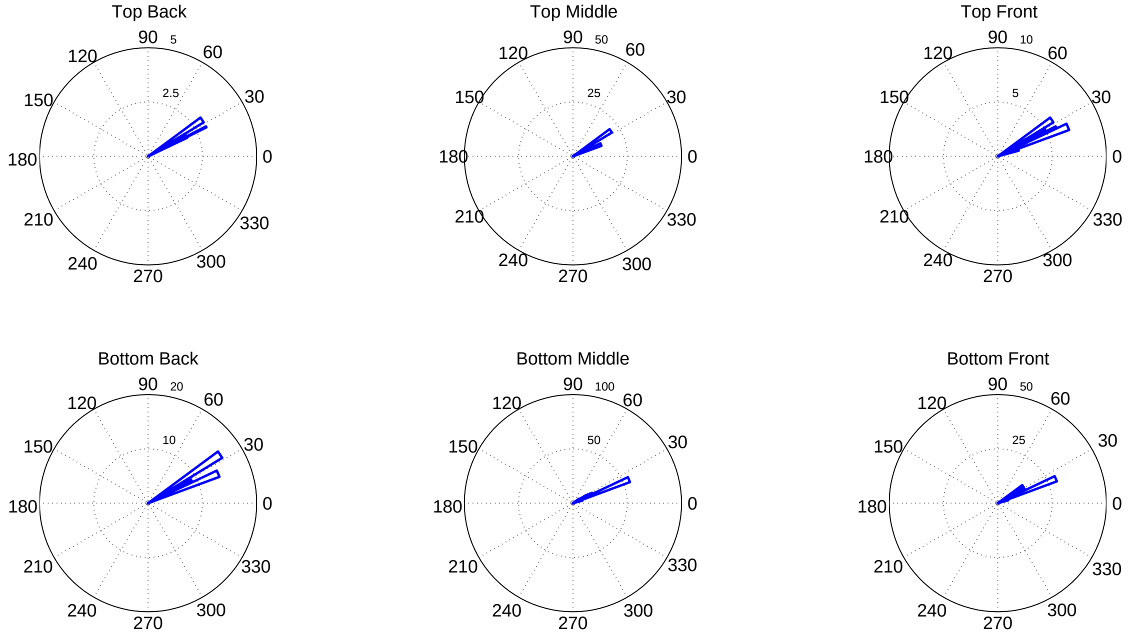
Segment	Variance s^2 (Active)	Variance s^2 (Passive)
Top Back	0.0096	0.0012
Bottom Back	0.2298	0.0037
Top Middle	0.0116	0.0019
Bottom Middle	0.4814	0.0017
Top Front	0.0289	0.0032
Bottom Front	0.5972	0.0019

7.5. Experiments: Directional dissent

In the experiments on directional dissent we will test to what extent active guidance via streaking is applicable to swarms that contain informed bees with different directional preferences. As pointed out already some species of honeybees can lift off before the selection process has converged towards a single nest-site. This seems especially to be the case for open-nesting species such as *A. florea*, which exhibits a very fuzzy nest-site selection process (MAKINSON ET AL., 2011). If the streaking hypothesis is a universal swarm guidance mechanism in bees, then it should be able to cope with directional dissent among informed individuals.


Top Back
Top Middle
Top Front
Bottom Back
Bottom Middle
Bottom Front

(a) Angular histogram under active guidance


Top Back
Top Middle
Top Front
Bottom Back
Bottom Middle
Bottom Front

(b) Angular histogram under passive guidance

Figure 7.5.: Angular histogram of the observed individual orientations in a swarm under differing guidance mechanisms at simulation-step 1000. The small numbers in the histograms correspond to the number of occurences.

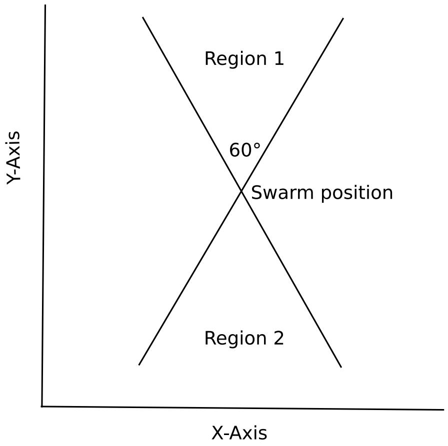


Figure 7.6.: Schematic of the spatial domain used for decision-making simulations.

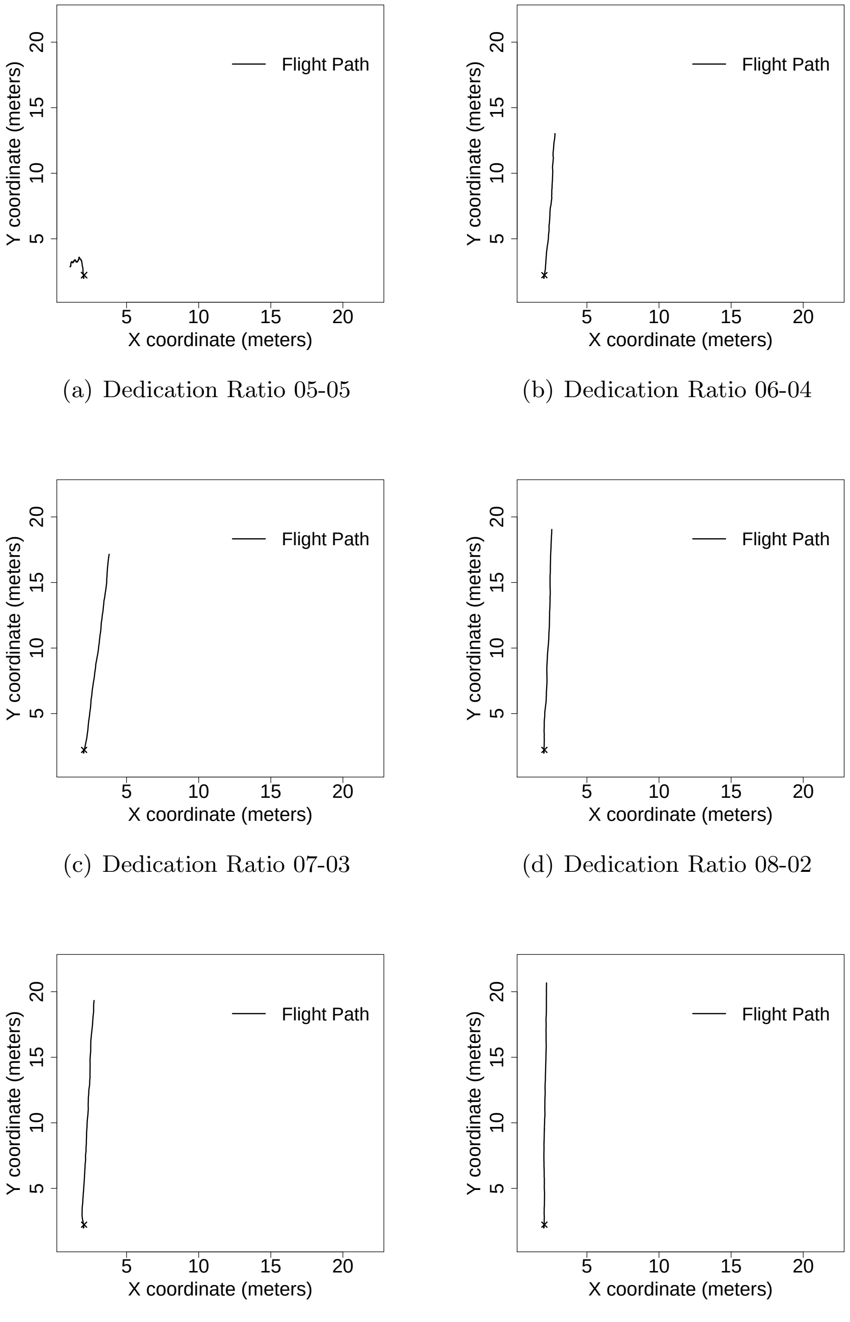
7.5.1. Directional dissent for streaking

To test a swarm's behaviour under directional dissent among streakers, we presented our model swarm with two large potential nesting regions (see Figure 7.6 for a sketch of the setup). Each region had an angular extent of 60°. We then varied the proportion of the streakers dedicated to each region, where each streaker chose a random direction within its region, representing its preferred flight direction.

As pointed out in Section 7.3 a smaller number of streakers was used here (i.e., 2% which corresponds to 10 individuals given a total swarm size of 500), as this fits the experimental data on open-nesting honeybees (Makinson et al., 2011). Thus a total of 6 simulation setups varying from total consensus (i.e., each streaker being dedicated to the same region) to dissent (half of the streakers attempting to steer the swarm to one region; the other half to the other region) were investigated. The exact direction of a streaker, within a region, was chosen at random and kept fixed during the 5 repeats.

Figure 7.7 depicts the flight paths of swarms with different levels of directional dissent. When the number of streakers dedicated to each region is equal, the swarm is not able to move in the direction of either region. Any attempt to move towards a specific region will be opposed by a strong force trying to guide the swarm in the other direction. Despite this strong directional conflict within such a swarm, a 'swarm breakup' (i.e., the division of the swarm into two sub-swarms that move towards one of the respective regions), does not happen, but the swarms showed a trend to move in the averaged direction of the two regions.

When the majority of streakers is dedicated towards the same region (here region 1), the swarm is able to move towards that region. This process can be thought of as an in-flight quorum, meaning that even if different regional preferences are present within the



(a) Dedication Ratio 05-05 (b) Dedication Ratio 06-04

(c) Dedication Ratio 07-03 (d) Dedication Ratio 08-02

(e) Dedication Ratio 09-01 (f) Dedication Ratio 10-00

Figure 7.7.: Average flightpath over 5 simulation runs of swarms under different dedication ratios x-y for region 1 and 2, where x represents the number of individuals dedicated to region 1 and y the ones dedicated to region 2. Crosses indicate the position of the swarm at the end of the initialization phase.

streaker population at the start of the flight, the swarm will be able to move towards the region the majority favours without an explicit quorum prior to lift-off.

Even though the swarm is able to fly towards the region favoured by the majority, directional dissent comes at a cost. As can be seen in Figure 7.7, the number of renegades that favour another region other than that of the majority impacts upon the distance the swarm is able to travel during one simulation run. Figure 7.8(a) depicts the distance between the initialization point of the swarm and the point it reached at the end of the simulation. The distance travelled by the swarm positively correlates with directional consent. Under strong directional dissent a swarm is only able to move ≈ 2 meters away from its starting position, which constitutes a tenth of the distance a swarm exhibiting total directional consent is able to travel. The effect of directional dissent is also reflected in the swarm's average flight speed. Figure 7.8(b) shows a swarm's average flight speed under the different levels of directional agreement. Again, in the presence of directional consent, the swarm flies fastest.

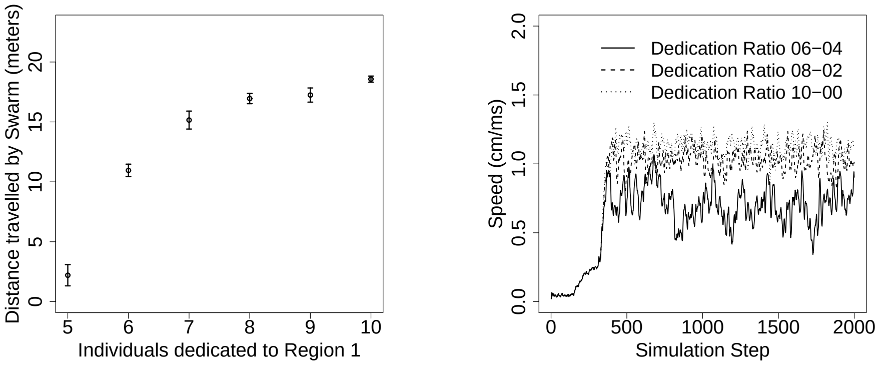
The effect of directional disagreement on the swarm's flight speed is not surprising given that the streaking mechanism depends on speed and direction propagating through the swarm. In other words, individuals that chase after streakers will be chased after by other (i.e., slower) individuals, creating a directed movement towards a given direction. If a swarm additionally contains opposing streakers, the directional consent in a swarm will get even smaller, as even if an individual will start chasing after a streaker it is very likely that it will slow down again, due to the opposing force of other streakers.

7.5.2. Streaking in A. florea

The results presented in the last section indicate that streaking is indeed applicable to swarms that contain informed individuals with differing directional preferences. In this final experiment the active guidance model was tested in more realistic conditions using experimental data from a previously studied *A. florea* swarm (i.e., MAKINSON ET AL. (2011) test swarm 1).

Figure 7.9(a) depicts the angular histogram of the observed dance directions in the last hour before lift-off of MAKINSON ET AL.'s (2011) test swarm 1. As outlined earlier, following MAKINSON ET AL. it is assumed that the individuals that dance in the hour before lift-off are responsible for the guidance of the swarm, which corresponds to 28 informed individuals.

Figure 7.9(b) shows the resulting average flightpath of a swarm with 2700 individuals. The simulated swarm flew in an average direction of 260.9° (std 0.6319°), which is close to the averaged dance direction of all dances observed one hour before lift-off (256°).



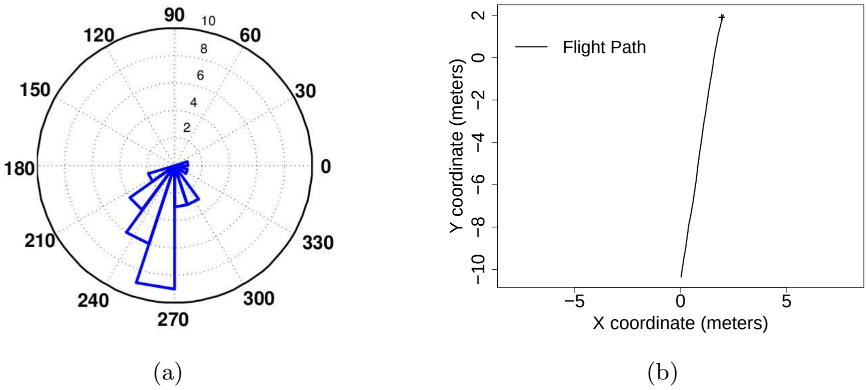
(a) (b)

Figure 7.8.: (a) Average distance between start and end position of swarms under different directional dissent. The x axis represents the number of individuals dedicated to region 1, the y axis the covered distance. Error bars show standard deviation; (b) Average swarm velocity development in the time course of a simulation under different dedication ratios (06 − 04,08 − 02,10 − 00).

Not only is active guidance able to cope with directional dissent among streakers, but it also enables a swarm to fly into a direction which constitutes the average direction (given that the directional preference is not completely opposing as in the experiment presented in the last section). As the swarm converges towards the average flight direction while en route, this can be seen as some sort of in-flight quorum. Swarms of different sizes did show some difference in flight speed, with smaller swarms moving faster than a large swarm (small swarms moved at a speed of ($\approx 1.2m/s$) while the large swarm moved at ($\approx 0.8m/s$)). These differences can be eliminated by parameter tuning, however because of the associated computational costs this was not done here. The main reason for running these simulations using a more realistic swarm size was to investigate if our small swarms showed aberrant behaviour. Clearly they do not.

7.6. Conclusion

This chapter investigated group guidance in the context of migrating honeybee swarms. A swarm flight towards a new nest-site constitutes the last step in the process of migration in honeybees, where a small number of informed individuals needs to lead the swarm towards a new home. To date, the empirically best supported guidance hypothesis is the vision hypothesis, which states that honeybee swarms are guided by fast-flying individuals (called streakers). Streakers repeatedly fly through the upper segment of a moving swarm, while uninformed individuals will chase after close-by individuals that are faster than themselves,



(a)

(b)

Figure 7.9.: (a) Angular histogram of the observed dance directions in test swarm 1 of Makinson et al. (2011) one hour before lift-off; (b) average flightpath of test swarm 1 over 5 simulation runs; Cross indicates the position of the swarm at the end of the initialization phase.

which sets the swarm en route. While some alternative honeybee guidance hypotheses such as the olifactory hypothesis have been experimentally rejected (Beekman et al., 2006), it is still debated whether honeybees are guided actively (via streakers) or in a passive fashion. Models of passive guidance have been used previously to simulate the group movement of fish (Ioannou et al., 2011) and assume that an individual is not able to distinguish between guide and non-guide. Instead, some individuals exhibit a directional preference, which influences movement behaviour. As a swarm will stick together these informed individuals are then able to guide the swarm in their desired direction. The advantage of passive guidance is that it is (from a behavioural perspective) simpler as it can be implemented via homogeneous individual behaviour, while active guidance requires two distinct behavioural classes.

To investigate which of these two rivalling hypotheses better captures honeybee swarm flight characteristics, a previously introduced model for group guidance in honeybees (Janson et al., 2005) was used. The original model was modified and extended to allow the study of both mechanisms under a nearly identical parameter set and used to investigate the extent to which the two guiding strategies are able to reproduce swarm characteristics from recent empirical findings on honeybee guidance (Schultz et al., 2008).

While both guiding mechanisms are able to guide the swarm in the right direction, our results show that active guidance better reflects the characteristics found in honeybee swarms than passive guidance. Schultz et al.'s (2008) empirical data suggests that moving honeybees are very noisy in terms of flight speed and directional orientation of individuals within the swarm. Under passive guidance these features can not be reproduced. This is due to the fact that individuals in a passively guided swarm are not able to distinguish between leaders and other uninformed individuals, which will lead per se to a stronger alignment within the group than when an individual is able to distinguish and takes only the guide into account. Active guidance is able to create such a diversity within a moving swarm, however to achieve empirical-like characteristics, individual bees have to be much more wayward than assumed in previous models (i.e., any information from surrounding individuals except faster flying ones is disregarded). Nevertheless, an actively guided swarm is still able to move in the desired direction and also shows more precision in terms of the swarm's flight directionality.

The second aspect investigated in this chapter concerns active guidance under directional dissent. Previous models on passive guidance (e.g., COUZIN ET AL. 2005) have shown that even in the absence of directional consensus, groups can still be guided by a small number of knowledgeable individuals. Here we demonstrated that group movement under directional dissent can also be achieved using the streaking mechanism. This suggests that streaking behaviour can be seen as a universal guidance mechanism in honeybee collective movement and not only a specialized form of guidance in specific honeybee species. However, we have also shown that when the direction of travel is decided while on the move, successful guidance is only possible if the group does not need to move to a specific location. If the exact location is vital, groups should only initiate movement once the specific direction of travel has been agreed upon. Hence, idiosyncrasies of the biological system have a huge influence on how groups of animals are guided.

8. Hunting the optimum: Honeybee nest-site selection as an optimization process

Bee-inspired algorithms are a new type of algorithm that has emerged in the field of swarm intelligence in recent years. These algorithms attempt to utilize the principles underlying the collective behaviour of honeybees, and have already been applied to various domains such as robotics (Gordon et al., 2003; Schmickl and Crailsheim, 2008), network routing (Farooq, 2008; Nakrani and Tovey, 2004), multi-agent systems (Lemmens et al., 2008) and optimization (Karaboga and Akay, 2009).

Although these algorithms draw their inspiration from honeybee behaviour, they are based on different concepts. In general, one can distinguish between two main classes of honeybee optimization algorithms: algorithms that utilize genetic and behavioural mechanisms underlying the bee's mating behaviour, and algorithms that take their inspiration from the bee's foraging behaviour. The biological background of these behaviours has been introduced in Chapter 5.

The first class of optimization algorithms makes use of the fact that a honeybee colony comprises a large number of individuals that are genetically heterogeneous due to the queen mating with multiple males. Many of the mating-inspired algorithms extend existing optimization algorithms from the field of evolutionary computation (Eiben and Smith, 2003) by introducing bee-inspired operators for mutation or crossover. Other algorithms in this class evolve populations of solutions by imitating a bee's maiden flight.

For the second class of optimization algorithms, foraging in honeybees is interesting as the underlying decentralized decision-making processes enable a colony to balance exploitation of known food sources with exploration for new and potentially better food sources in a dynamic environment (Beekman et al., 2007). Algorithms based on foraging usually use artificial bees to search for solutions and thus associate solutions with food sources. Depending on the number of food sources (solutions) found and their quality, a subset of the artificial bee population will explore the environment (search space) by finding new food sources (creating new solutions), while the remaining bees exploit the environment around the found food sources in order to try and improve current food sources (i.e., they perform local search operations in order to improve the current solutions).

In this chapter we introduce a third possible class of optimization algorithms based on nest-site selection. In contrast to foraging, where bees can typically forage at different locations simultaneously, nest-site selection always involves the selection of a single new site. Finding a good nest site is vital for the continued survival of the colony and the corresponding decision-making process should be flexible enough to allow the discovery of superior new nest sites during later stages of the selection process. This makes nest-site selection of particular interest for dynamic optimization problems, in which the problem instance is likely to change during the optimization process.

Based on the spatial nest-site selection model used in Chapter 6 to investigate the impact of nest-site distribution in the environment on the decision-making process, we will assess the optimization potential of this behaviour. Additionally an optimization scheme, called the bee nest-site selection scheme (BNSSS), based on nest-site selection is introduced. A realization of this scheme, the BeeNest algorithm is outlined and applied to the domain of molecular docking.

8.1. Related work: Current bee-inspired algorithms

As outlined above, current bee-inspired optimization approaches are based on one of two behaviours found in bees: foraging or mating. This section outlines current algorithms based on these behaviours. For previous reviews of bee optimization methods and related techniques, the interested reader should refer to Karaboga and Akay (2009).

8.1.1. Mating-based optimization algorithms

Honeybee Mating Optimization Algorithm One example of a genetically-based optimization algorithm is the honeybee mating optimization algorithm (HBMO) developed by Abbass (2001a,b) for discrete optimization problems. The HBMO algorithm contains four main (artificial) entities called queens, drones, brood, and workers. The algorithm operates in two stages: maiden flight and brood development. Both stages are executed alternately until a stopping criterion is satisfied.

The HBMO operates with several queens. At the beginning of the maiden flight each queen is equipped with a single randomly generated reference solution. In addition a queen has a flight speed s, an energy e, and a limit for the amount of sperm (i.e., sample solutions from drones) she can store in her spermatheca (i.e., a pool of new sample solutions). A queen stops her maiden flight either when her energy is depleted or when a maximum number of drone solutions has been collected. During each step of her maiden flight a queen Q encounters a drone D. She will absorb the drone's solution with a probability of

$$p(Q, D) = e^{-d/s} \tag{8.1}$$

where d represents the fitness difference between the queen's reference solution and the drone's solution and s represents the flight speed of the queen. As can be seen from $p(Q, D)$ a queen is very likely to accept a drone's solution if the solution is better than the queen's reference solution or if her speed is high. After each step the queen's flight speed and energy are decreased, which results in the queen becoming more selective with respect to absorbing potential drone solutions over the course of her flight.

The maiden flight is completed when a maximum number of drone solutions has been collected, or when the queen's energy is depleted. At this point, a queen will mate with a drone solution randomly selected from those in her spermatheca. Mating involves the application of a crossover operation to the selected drone solution and the queen's reference solutions, and results in a single offspring. In addition a mutation operation might be applied. The survival of the offspring depends on the quality of this offspring reference solution.

The offspring solutions of the queens are nursed by the workers during the brood development stage. A worker represents a local search heuristic and nursing corresponds to the application of this heuristic to try and improve the offspring solution. Then, before a new maiden flight stage is started the least fit queens are replaced with the fittest offspring until no offspring is fitter than the least fit queen. Again, this process is repeated until a stopping criterion is satisfied.

Initially the HBMO algorithm was proposed for solving the Boolean satisfiability problem (ABBASS, 2001a,b) and has since been adapted for several other problems such as water reservoir management (AFSHAR ET AL., 2007; HADDAD ET AL., 2008; MOHAN AND BABU, 2010), data clustering (FATHIAN AND AMIRI, 2008; MARINAKIS ET AL., 2008) and vehicle routing (MARINAKIS ET AL., 2009a).

Bumble bees mating optimization The Bumble Bees Mating Optimization algorithm (BBMO) was introduced by MARINAKIS ET AL. (2009b) and is closely related to HBMO. Here the mating behaviour of bumble bees is used as a template for optimization. The algorithm is initialized by creating a number of solutions to a given problem, with each solution corresponding to the genotype of a bumblebee. The best solution to the initial population becomes the queen, while the rest of the population are considered to be drones. As in the HBMO algorithm the queen will select drones in terms of fitness and mate with them until her spermatheca is full.

After this initial mating the queen will start to lay eggs. A queen is able to produce two kinds of bees: workers and drones. Workers are the result of a point-wise crossover between the queen's genotype and one of the drones (i.e., given a certain crossover rate C_r it is decided for each position in the genome if the queen's genetic information is kept $rand(0,1) < C_r$ or the drone's genetic information will be used). In contrast, drones are created via random mutations of the queens genotype.

The fittest individuals of the worker population are considered as new queen candidates. Via a feeding process (which corresponds to local search) the new queens are fed by the old queen as well as a number of workers. The local search is applied multiple times to single positions of the candidate queens genotype (i.e., single dimensions of the problem) and realized via the following equation:

$$\begin{aligned} nq_i = nq_i + (b_{max} - \frac{(b_{max} - b_{min}) \cdot lsi}{lsi_{max}} \cdot (nq_i - q) + \\ \frac{1}{M} \cdot \sum_{k=1}^{M} (b_{min} - \frac{(b_{max} - b_{min}) \cdot lsi}{lsi_{max}} \cdot (nq_i - w_k) \end{aligned} \tag{8.2}$$

Here nq_i corresponds to the genotype of a new queen candidate, q denotes the solution of the old queen and w_k corresponds to a worker bee. M denotes the number of workers that will feed the candidate queen. The impact of the workers versus the impact of the old queen on the local search is steered via the parameters b_{max}, b_{min}, lsi and lsi_{max}, where lsi_{max} corresponds to the maximum number of local search steps which are performed, lsi corresponds to the current local search step and b_{max} and b_{min} are parameters in the range of (0,1). Marinakis et al. choose b_{max} to be close to 1 and b_{min} to be close to 0, which leads to the local search incorporating the old queen's solution in early local search iterations, while focusing on the workers in the later phase.

After the feeding the candidates, queens as well as the drones leave the hive and mate and the fittest fertilized candidate queens survive and continue the reproduction process.

The BBMO algorithm was initially introduced as a hybrid approach to clustering (Marinakis et al., 2009b) and has since been applied to the vehicle routing problem (Marinakis and Marinaki, 2010) as well as unconstrained optimization problems (Marinakis et al., 2010).

Honeybee inspired evolutionary computation Other approaches that are based on honeybee mating utilise bee-inspired operators within existing evolutionary computation algorithms, see for example Sato and Hagiwara's bee system (Sato and Hagiwara, 1997), Jung's queen-bee evolution (Jung, 2003) or Karci's bee-inspired genetic crossover operator (Karci, 2004). As these methods extend well-known optimization methods we will not go into further detail here.

8.1.2. Foraging behaviour based approaches

Foraging behaviour based approaches take inspiration from the mechanisms underlying the foraging process in honeybees. Besides the experimental studies outlined in Chapter 5, several theoretical models support and outline the effectiveness of the honeybee's decentralized decision-making process when foraging (Beekman and Lew, 2008; de Vries and Biesmeijer, 1998; Dornhaus et al., 2006; Seeley et al., 1991; Sherman and Visscher, 2002; Sumpter and Pratt, 2003).

Sherman and Visscher (2002) investigated when waggle-dance recruitment is beneficial. Their results suggest that this recruitment increases the amount of food a colony can collect when resources are scarce. A recent study by Dornhaus et al. (2006) suggests that the recruitment dance is especially beneficial if resources are few in number and of variable quality. Beekman and Lew (2008) found that recruitment is most beneficial if the average success in locating new food patches falls below the average success of recruitment. Additionally, they showed that communication facilitates the rapid exploitation of highly profitable food sources when several food sources of different quality are present. Thus, the bees' dance communication regulates the trade-off between exploitation and exploration.

These studies underline the usefulness of honeybee foraging behaviour in terms of optimization in a dynamic environment in which resources are sparse and differ in quality, as is the case in many problem domains of optimization. Moreover, the above-mentioned studies outline the importance of direct communication between the bees. Inspired by these findings, direct information transfer plays an important role in the algorithms which are outlined below. This is in contrast to ant colony optimization algorithms that rely on indirect communication via artificial pheromones (Bonabeau et al., 1999).

The Artificial Bee Colony Algorithm (ABC)) The Artificial Bee Colony algorithm (ABC) was introduced by Karaboga (Karaboga, 2005; Karaboga and Basturk, 2007b) for function optimization. Each solution (i.e., a position in the search space) represents a potential food patch and the solution quality corresponds to the food patch's quality. Agents (artificial bees) search and exploit the food sources in search space.

The ABC uses three types of agents: employed bees, onlooker bees, and scouts. Employed bees (EB) are associated with the current solutions of the algorithm. In every step of the algorithm an EB tries to improve the solution it represents using a local search step, after which it will try to recruit onlooker bees (OBs) for its current position. OBs select among the promoted positions according to their quality, meaning that better solutions will attract more OBs. Once an OB has selected an EB and thus a solution it tries to optimize the EB's position by means of a local search step. An EB updates its position if an OB it recruited was able to spot a better position, otherwise it remains at its current position. In addition, an EB will abandon its position if it was not able to improve its position for a certain number of steps. When an EB abandons its position it becomes a scout, meaning that it selects a random position in the search space and becomes employed at that position.

The algorithm can be described in more detail as follows: given a *dim* dimensional function F and a population of n agents, $n_e = n/2$ EBs and $n_o = n/2$ OBs. The algorithm is initialized by placing EB i ($i \in n_e$) on a random location θ_i in the search space. $F(\theta_i)$ is then the quality of the position of EB i.

In every iteration, each EB tries to improve its location using a local search step. First, EB i calculates a new a candidate solution

$$\theta_i^* = \theta_i + rand(-1,1) \cdot (\theta_i - \theta_k) \tag{8.3}$$

where θ_k corresponds to the position of another randomly chosen EB with index k ($i \neq k$) and $rand(-1,1)$ constitutes a random number between -1 and 1 drawn from a uniform distribution. Note that formula 8.3 is typically not applied for all dimensions of θ_i. While the number of dimensions that are taken into account in the case of a constraint optimization problem depends on a parameter called the perturbation rate (see KARABOGA AND BASTURK (2007a) for more details), only one dimension is taken into account for unconstrained optimization problems. The dimension(s) to be altered are randomly chosen. After a new candidate solution is calculated a greedy selection mechanism is used in order to decide if θ_i should be discarded

$$\theta_i = \begin{cases} \theta_i & \text{if } F(\theta_i) > F(\theta_i^*) \\ \theta_i^* & \text{else} \end{cases} \tag{8.4}$$

After each EB has updated its position, each OB chooses one of the current solutions. A standard roulette wheel selection (EIBEN AND SMITH, 2003).

$$P_i = \frac{F(\theta_i)}{\sum_{k=1}^{n_e} F(\theta_k)} \tag{8.5}$$

is used, and better solutions attract more OBs. After choosing a solution an OB tries to improve the solution using the same mechanism as outlined in Eq. 8.3. The EB that corresponds to this solution updates its position if a better position is found by the OB.

The algorithm keeps track of how many steps an EB has been at the same solution. If the number of steps spent on the same position reaches a certain value *limit* the EB abandons its position and scouts for a new position, which corresponds to choosing a random position

Algorithm 2 Artificial Bee Colony

```
place each employed bee on a random position in the search space
while stopping criterion not met do
    for all EBs do
        if steps on same position == limit then
            choose random position in search space
        else
            try improve position (according to Eq. 8.3)
            if better position found then
                change position
                reset steps on same position
            end if
        end if
    end for
    for all OBs do
        choose position of employed bee (according to Eq. 8.8)
        try improve position (according to Eq. 8.3)
    end for
end while
```

in search space. This parameter thus controls the exploitation/exploration rate of the system. In KARABOGA AND BASTURK (2008) the impact *limit* was investigated, and found to depend on the problem's dimensionality and the number of employed bees in the system, with an optimal value given as $limit = n_e \cdot dim$. In a recent study (AKAY AND KARABOGA, 2009) this suggestion was re-examined. It was concluded that small colonies should use a value $limit > n_e \cdot dim$, as they need more time to search in the vicinity of the EBs' solutions than large colonies. In a very recent study (DIWOLD ET AL., 2011a) the influence of ABC's parameters on its optimization behaviour was investigated, including the influence of the OBs on the algorithm's performance, showing that the ideal parameter values depend on the hardness of the optimization goal and that the standard values suggested in the literature should be applied with care. DIWOLD ET AL. were also able to show that using OBs is not always of advantage and that the ABC's performance decreases when used for problems where the optimum is not located in the centre of the search space. Additionally, two new selection schemes were introduced which significantly improved the ABC's performance. For a better understanding, the basic ABC algorithm is outlined in Algorithm 2.

The ABC has been used in several problem domains such as unconstrained (KARABOGA AND BASTURK, 2007b, 2008) and constrained numerical optimization (KARABOGA AND BASTURK, 2007a), data clustering (KARABOGA AND OZTURK, 2011), the training of neuronal networks (Karaboga et al., 2007) and protein structure prediction (Bahamish et al., 2009).

Bees Algorithm (BA) The Bees Algorithm (BA) was introduced by Pham et al. (2006b) as an optimization method for continuous and combinatorial function optimization.

As in the ABC, the population of bees is divided into two groups: scouts and recruits. While scouts are responsible for the exploration of the search space the recruits try to exploit (i.e., improve) found solutions via local search. The algorithm depends on a set of parameters which will be outlined briefly below.

The optimization process starts by assigning each of n scout bees to a random position in the search space. A scout's fitness corresponds to the quality of the position (solution) it currently occupies. The best $m \leq n$ scouts are selected and the rest are discarded (selected scouts are referred to as selected bees). The selected bees are further partitioned according to their fitness into e elite selected bees and the $m - e$ non-elite selected bees.

Each selected bee is assigned a number of recruits, and how many depends on the solution quality of the bee. Each elite bee receives neq recruits, each non-elite bee nsp recruits.

Each recruit performs a local search step at its assigned position according to

$$x_j^* = (x_j - ngh) + (rand(0,1) \cdot ngh \cdot 2) \tag{8.6}$$

with ngh denoting the search patch size. The best improvement of a selected bee's solution will replace this solution. If none of the solutions found by the recruits yields an improvement over the selected bee's solution, the solution is maintained. The scout population is filled up with these m solutions and $n - m$ random solutions and the algorithm repeats until a stop criterion is satisfied. It should be noted that the BA algorithm was recently improved (Pham et al., 2008) by introducing more local search methods such as mutation, creep, crossover, interpolation and extrapolation, that can be used by recruits to improve given selected solutions. The algorithm underlying a standard BA is outlined in Algorithm 3 according to Pham et al. (2006b).

The BA has been applied to various engineering problems, such as the training of neural networks (Pham et al., 2006a,b,c,d), controller formation (Pham et al., 2009), image analysis (Olague and Puente, 2006), job multi-objective optimization (Pham and Ghanbarzadeh, 2007) and data traffic load balancing (Bernardino et al., 2011).

Bee Colony Optimization algorithm (BCO) The Bee Colony Optimization algorithm (BCO) (Teodorovic and Dell'Orco, 2005) constitutes a generalized and improved

Algorithm 3 Bees Algorithm

```
place each bee on a random position in the search space
evaluate the fitness of the population
while stop criterion not satisfied do
    select solutions for a local search (exploitation)
    assign bees to commit local search on selected solutions and evaluate fitness
    for each solution select the best improvement
    replace remaining solutions with random solutions (scout)
end while
```

version of the Bee System algorithm (Lucic and Teodorovic, 2001). Both algorithms were designed to tackle combinatorial optimization problems. As the two algorithms are basically identical, we will treat them as one in the following.

BCO divides the optimization process into $I \geq 1$ iterations, where I is a parameter set by the user. During each iteration, B virtual bees try to construct a solution for the given problem. Due to the combinatorial nature of the problems BCO tackles, solutions are constructed as a consecutive extension of initial partial solutions. To do so each iteration is divided into a finite sequence of $m \geq 1$ stages $S = \{s_1, s_2, \ldots, s_m\}$.

During a stage s_j a bee will extend its current partial solution by adding an available partial solution. In the BCO terminology extending a current solution with a partial solution is called the forward pass. How a forward pass is implemented depends on the underlying problem. In Lucic and Teodorovic (2001) the BCO (then called Bee System) was used to solve the travelling salesperson problem, and the Logit model (Cramer, 2003) was used to decide how to extend partial solutions.

After each bee has performed a forward pass, a backward pass is performed, in which all bees compare their current partial solutions. On the basis of this comparison bees decide whether or not to keep their current partial solution, promote it to other bees, or abandon it. Bees that give up their current partial solution will choose one of the solutions promoted by other bees. The backward pass ends a stage. The sequence of stages leads to an iterative solution build-up where bad partial solutions will be abandoned and the search will focus on promising partial solutions.

At the end of each iteration, it is determined whether the best solution found in that iteration should become the new global best solution. The underlying algorithm is outlined in Algorithm 4.

The BCO has been used to solve problems in traffic and transportation (Lucic and Teodorovic, 2001, 2002, 2003; Teodorovic and Dell'Orco, 2005; Teodorovic and Dell'Orco, 2008; Teodorovic et al., 2006).

Algorithm 4 Bee Colony Optimization

```
initialization
for all $I$ iterations do
    for all $m$ stages do
        for all $B$ bees do
            forward pass: choose partial solution
        end for
        for all $B$ bees do
            backward pass: exchange information about partial solutions with bees in nest
        end for
    end for
    if best solution obtained in iteration is global best, update best-known solution
end for
```

The Bee Colony-Inspired Algorithm (BCiA) The bee colony-inspired algorithm (BCiA) was recently introduced by Häckel and Dippold (2009) for the vehicle routing problem with time windows (VRPWTW). Given a number of customers that have to be supplied with goods within a certain time window, optimizing the VRPWTW requires finding a route schedule that minimizes the associated costs (number of vehicles needed and total tour length). In order to avoid conflicts between the two optimization objectives, BCiA operates in two stages – in the first it tries to reduce the number of vehicles needed for a valid solution and in the second it tries to minimize the total tour length. Instead of a single population of virtual bees, BCiA uses 2 populations P_1 and P_2 operating in stages 1 and 2, respectively. The principles used in BCiA are similar to those used in the ABC (Karaboga and Basturk, 2007b), but ABC was designed for numerical optimization problems whereas BCiA tackles discrete optimization problems.

BCiA uses virtual bee populations that consist of three bee types: employed bees (EBs), follower bees (FBs), and scouts. The separation of roles within a population is similar to ABC: EBs are associated to current solutions, FBs try to improve those solutions (similar to OBs in ABC), and scouts provide the population with new solutions (this is done by EBs in ABC).

The two populations of BCiA each consist of n_{eb} EBs, n_{fb} FBs and n_{scout} scouts. Initially, the EBs of both populations are initialized with random solutions. Each iteration of the algorithm is divided into two stages. In the first stage the first population P_1 tries to improve its solutions. This is done similarly to the ABC, i.e., each FB chooses an EB based on its fitness with respect to the first optimization goal F_1.

FBs then try to improve an EB's solution by constructing a new solution. During the construction process a new solution is constructed taking the EB's solution into account. The details of the process depend on the specific optimization problem and will not be discussed here (the interested reader is referred to the publication itself Häckel and Dippold (2009)). If the solution found by an FB has better quality than the EB's current solution, the latter is replaced by the FB's solution.

Algorithm 5 The bee colony-inspired algorithm (BCiA)

```
initialize populations
while stop criteria not met do
    for all i $\in \{1, 2\}$ do
        for all FBs $\in P_i$ do
            choose EB $\in P_i$
            construct new solution regarding $F_i$ using $EB$
        end for
        update EBs $\in P_i$ according to the solutions found by the FBs
        for all scouts $\in P_i$ do
            construct a new solution with respect to $F_i$
            exchange worst EB if better solution is found by scout
        end for
        if i equals 1 then
            update EBS in $P_2$ according to $P_1$
        else
            update EBS in $P_1$ according to $P_2$
        end if
    end for
    check age of solutions and replace them if age exceeds limit
end while
```

After the FBs try to improve the current solutions in the populations, the scouts create new solutions. Scouts do not use a reference solution when generating a new solution, but apart from this the generation process is identical to the one used by the FBs. The best scout solutions will replace the worst EBs if their quality is better. After the improvement step a solution exchange between the two populations P_1 and P_2 is initiated. In stage 1 each EB in P_1 that is not yet present in P_2 and has a better quality regarding the optimization goal of P_2 (i.e., F_2) than the worst EB in population P_2 is added to P_2, while the worst EB in P_2 is deleted. The converse occurs in stage 2.

The second stage of the iteration is then executed. It follows the same sequence of actions as in the first stage but uses population P_2 and the optimization goal F_2. At the end of each iteration of stage 2 the age (i.e., number of iterations the solution was not improved) of all solutions (i.e., EBs) is checked. Solutions that exceed a certain age are exchanged (similar to ABC). Any old solutions of the population P_1 are substituted by new scout-generated solutions, and if any solutions in P_2 are abandoned, they are substituted by the best solution with respect to F_2 from population P_1. The substitution in P_2 only

Table 8.1.: Look-up Table for adjusting r_i according to the profitability rating.

Profitability Rating	r_i
$Pf_i < 0.9 \cdot Pf_{colony}$	0.60
$0.9 \cdot Pf_{colony} \leq Pf_i \leq 0.95 \cdot Pf_{colony}$	0.20
$0.95 \cdot Pf_{colony} \leq Pf_i \leq 1.15 \cdot Pf_{colony}$	0.02
$1.15 \cdot Pf_{colony} \leq Pf_i$	0.00

happens if the solution in P_1 contains the best (i.e., smallest) number of vehicles known so far, otherwise no substitution takes place.

BCiA terminates when a stop criterion is satisfied. The algorithm is outlined in algorithm 5.

Bee Colony Optimization Algorithm (BCOA) Introduced by Chong et al. in 2006, the Bee Colony Optimization Algorithm (BCOA) was originally proposed for the job shop scheduling problem (Chong et al., 2006, 2007).

BCOA consists of a population of n foragers. During each iteration each forager f_i constructs a solution for the given optimization problem. The foragers then promote their solutions to each other. Based on the quality of its own solution, a forager can decide to keep its previous solution or abandon it and adopt that of another forager. After each forager has decided, it will create a new solution based on its current solution. The general principle of BCOA has similarities to the BCO which was outlined above.

Each iteration in the BCOA can be divided into two phases: the dancing phase and the foraging phase. During an iteration each forager constructs a solution for the given problem (how will be explained later). Then each forager f_i ($i \in [1, n]$) returns to the hive and performs a waggle dance with a certain probability p. Let $Pf_i = 1/C^i_{max}$ denote the profitability rating of the solution a dancing forager f_i is trying to promote, where C^i_{max} represents the fitness of f_i's current solution. The average profitability rating of all dancing foragers is given by $Pf_{colony} = 1/n_d \sum_{i \in F_d} Pf_i$, with n_d corresponding to the number of dancing bees, and F_d the set of dancing bees.

The waggle dance of forager f_i will last for $D = d_i \cdot A$ steps with $d_i = Pf_i/Pf_{colony}$ depending on the profitability rating of the obtained solution (e.g., make span, tour length) and $0 < A$ denoting a waggle dance scaling factor. Each forager also attempts to follow a randomly selected dance of another forager with probability r_i, with r_i depending on the profitability rating of the solution found (see Table 8.1) (i.e., foragers that found a solution with high profitability rating are unlikely to follow another forager's dance).

The BCOA has been extended for the travelling salesman problem (Lu and Zhou, 2008; Wong et al., 2008, 2009) and a recent modification of the algorithm for feature selection problems has also been proposed (Subbotin and Oleinik, 2009).

The Virtual Bee Algorithm (VBA) The Virtual Bee Algorithm (VBA) algorithm scheme for numerical optimization was introduced by YANG (2005). It proposes function optimization via a set of virtual bees that are initialized on random positions in a given search space. Each position of the search space is assigned virtual food, such that the food quality corresponds to the value of the function to be optimized at a given position. Virtual bees will explore the search space and communicate found food patches to other bees. Bees that receive information about other food patches will incorporate this information into their search behaviour. Please note that Yang gives only a very schematic description of the algorithm. The exact details of how communication, search, and incorporation of solutions obtained from other individuals is handled is not explained in YANG (2005). The VBA was tested on two 2-dimensional test functions and the author claims that it outperformed a standard genetic algorithm. As the article lacks detailed information on the proposed algorithm it is hard to validate these findings.

8.2. Nest-site selection as an optimization process

To test the optimization potential of nest-site selection the spatial nest-site selection model introduced in Chapter 6 was used. Unless stated otherwise, the same parametrization as in Chapter 6 was used in the optimization experiments. All presented results are averaged over 10 runs and the number of bees was set to $n = 500$, as this resembles a reasonable number of real honeybees taking part in the selection process.

8.2.1. Experiment: Nest-site selection in a dynamic environment

This experiment tests how the nest-site selection process performs in an environment where the quality of the sites changes over time. In an environment with two nest sites located equally far from the swarm but which differ in quality, the number of scouts would build up quickly at the higher quality nest-site if the nest-sites quality remains the same. In this experiment however the quality of the nest sites is swapped at regular intervals. While such a situation is unlikely to occur in nature, changing optima are ubiquitous in dynamic optimization problems.

The environment contains two potential nest sites $n1$, $n2$ that are located in opposite directions 150 meters away from the swarm's position. Site $n1$ is initialized with a good quality ($q_{good} = 75$) while $n2$ is initialized with bad quality ($q_{bad} = 45$). The simulation runs for 32 hours, corresponding to 115200 simulation steps. At an interval of 28800 simulation steps (i.e., every 8 hours) the qualities of the nest sites are swapped. This leads to a total of 3 quality switches over the whole simulation run.

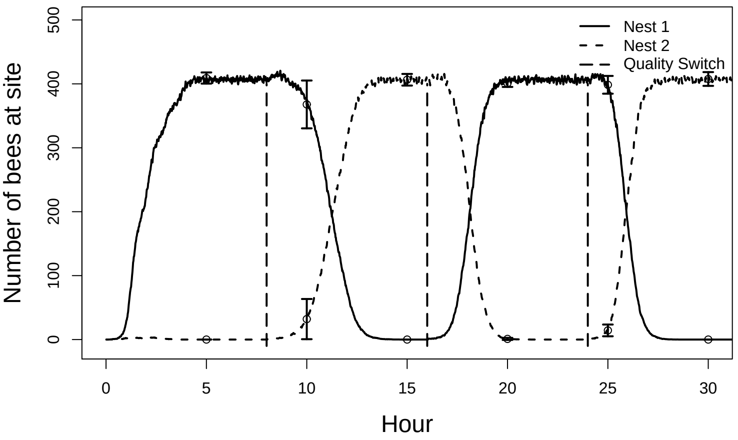


Figure 8.1.: Average number of bees assessing a nest site in a system where the site qualities are swapped every 8 hours (2880 simulation steps). Error bars show standard deviation.

Given that there is a probability that a scout will not find a given nest site, it is possible that the swarm discovers one nest site only during the simulation, or perhaps even none at all. In addition, a swarm can "forget" a low quality nest site if no dance for that site is sustained prior to the switch in nest-site quality. When this happens the site needs to be rediscovered after the qualities have been swapped. To ensure that bees are aware of both nest sites each time their quality is switched, a random bee is chosen that starts dancing for the nest site that was of low quality but switched to high quality.

Figure 8.1 depicts the average number of bees at each nest site over 10 simulation runs. The swarm quickly adapts to changes in nest-site quality. It is clear that the process is rather slow as it takes the swarm approximately 2 hours to adapt to the change in quality. However, this is not necessarily a disadvantage as it makes a swarm resilient towards noise. Even though in real bees the quality of a nest site is most likely to remain constant, the discovery of a new nest site also constitutes a change in the swarm's environment. Without the ability to react to changes in the environment, a swarm could get stuck in a suboptimal solution if it finds a nest site of mediocre quality early in the decision-making process. In terms of optimization, adapting to a dynamic environment is an interesting aspect, as it can be applied to the detection of changing locations of the optima in problems with dynamic fitness functions.

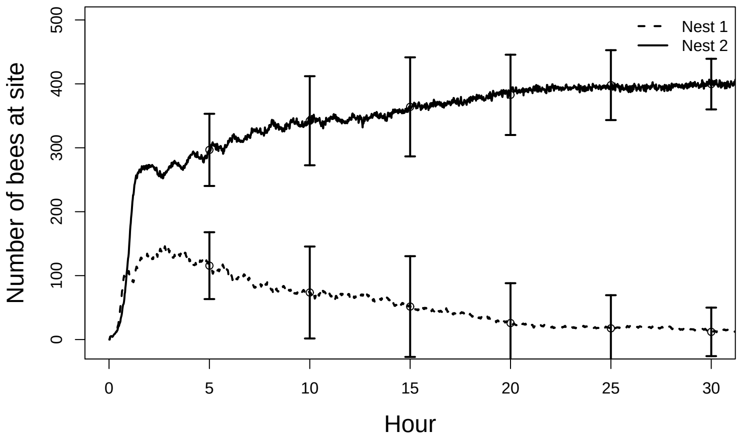


Figure 8.2.: Average number of bees assessing a nest-site when the nest-site of high quality changes in quality. The quality of nest site $n1$ changes each 1800 simulation steps between $q_{good} = 75$ and $q_{vbad} = 35$, whereas the quality of nest site $n2$ is constant at $q_{mediocre} = 55$. Error bars show standard deviation.

8.2.2. Experiment: Nest-site selection in a noisy environment

This experiment tests whether the swarm is capable of selecting a stable mediocre quality nest site and disregarding a site of sometimes high but very unstable quality.

The number of bees and the number and position of the potential nest-sites is the same as in Experiment 1, but the quality of nest-site $n2$ is kept constant at mediocre quality ($q_{mediocre} = 55$), whereas the quality of nest site $n1$ changes at an interval of 1800 simulation steps (i.e., every 30 minutes), alternating between good ($q_{good} = 75$) and very bad ($q_{vbad} = 35$). Nest-site $n1$ is initialized with a good quality q_{good}. A simulation run lasted for 32 hours corresponding to 115200 simulation steps. To ensure that the swarm is aware of each nest site, a random bee starts dancing for each nest site at the beginning of the simulation.

Figure 8.2 shows the average number of bees at the two nest sites over 10 simulation runs. As can be seen, the swarm is able to direct most scouts towards the stable mediocre nest site. At the start of a simulation the number of bees builds up quickly at both nest sites, which is caused by the fact that one bee starts dancing for each nest site when the simulation is started. However, as bees begin to revisit the nest sites, more bees are recruited towards the mediocre stable site. This is due to the revisit behaviour of honeybees. Even though many bees will initially promote nest-site $n1$ more strongly than nest-site $n2$ due to better quality, the ongoing revisiting will cause many bees to abandon the unstable site and choose the stable site. Nest-site $n1$ will never be completely

Table 8.2.: Test functions and domain space range (R). The dimension of each function is 2.

		R
Sphere	$f_{sp}(\vec{x}) = \sum_{i=1}^{n} x_i^2$	$[-25; 25]^n$
Booth	$f_{bt}(\vec{x}) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$	$[-10; 10]^n$

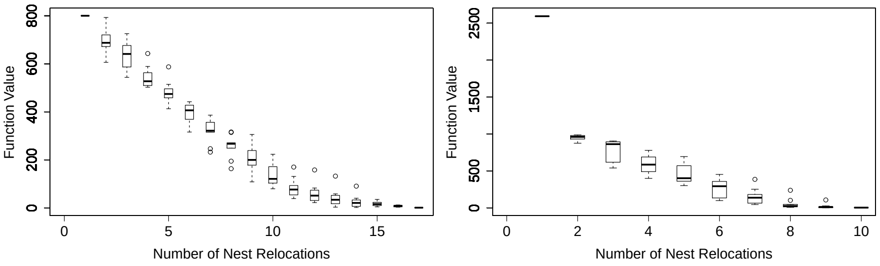
abandoned due to the fact that some visiting bees will always experience it as a very good nest site and thus revisit it. In general this experiment demonstrates that the nest-site selection mechanism is, to some extent, resilient towards noise.

8.2.3. Experiment: Function optimization via iterative nest-site selection

When searching for a new nest-site, bees typically have to decide between several discovered nest-sites. In case of the European honeybee *A. mellifera*, the number of possible good nest sites is limited as they live in cavities. The swarm needs to ensure that it decides for the best site possible so that it becomes unlikely that the nest site turns out to be of insufficient quality, forcing the swarm to move again. However, for bee species that live in the open such as the Dwarf honeybee *A. florea*, the quality of the nest site appears to be less important and the swarm has the chance to "upgrade" if its initial decision was suboptimal (OLDROYD ET AL., 2008).

Thus it interesting to see if an iterative nest-site selection process as found in *A. florea* can lead to an optimization in an environment with many potential nest sites. In this experiment it is assumed that the swarm's environment corresponds to the search space of a continuous function. Each position in the environment constitutes a potential nest site, and its quality corresponds to a value of the function at that position. The test functions used in the experiment and their associated parameter values are given in Table 8.2. The swarm is initially placed at position [-20,-20] for the Sphere function and [-10,-10] for the Booth function.

For this experiment the scouting behaviour of the bees has been changed, as the first version of the extended model is orientated to the behaviour of *A. mellifera* where a scout assesses a nest-site for a certain period of time before returning to the swarm. However in this experiment each location corresponds to a potential nest-site, and scouts would thus immediately start to assess sites after a single scouting step. To overcome this, a scout in this experiment remembers the best position it encountered during its scouting trip. If the quality of that position is better than the current location of the swarm it starts dancing for that site.



(a) Sphere

(b) Booth

Figure 8.3.: Boxplots of the quality of the occupied nest site over several relocations for the two test functions.

The quality of a newly discovered site is defined relative to the quality of the current location of the swarm. If a scout discovers a nest-site that is 60% better than the swarm's current location, this site is assigned quality 60.

Bees that followed a dance for a potential nest-site become recruits for this site and will fly towards it. If they encounter a better site on their way to this nest site, they abandon the recruitment process and become scouts. Recruits that do not find an advertised nest site also become scouts.

Nest-sites are assessed by recruits and returning bees for a certain amount of time, T_{ASSESS}, during which a bee counts the number of other bees present at the site. If the number of bees at a site reaches a given quorum $q = 10$ the swarm is relocated to this new site and the nest-site selection process is restarted. The parameter values used in this experiment are: step size $step = 0.1$, scouting $T_{SCOUT} = 100$, and assessment time $T_{ASSESS} = 20$. A simulation run is stopped when no swarm relocation occurs within 3600 simulation steps.

The changes in the quality of the found nest sites for both test functions over several nest-site relocations is depicted in Figure 8.3. As can be seen, the bees are able to iteratively optimize the position of the swarm within the search space. The optimization process is limited by several factors. Since scout time T_{SCOUT} and step size $step$ are fixed, scouts are only able to explore a certain range around the swarm's current location whereas a fixed step size prevents scouts from finding better solutions as they are likely to fly over them. This is especially the case when the swarm is close to the global optimum when scouts should actually search at a finer scale. Another limiting factor is the quality assessment. Remember that the quality of a newly found nest site is determined according to the potential improvement with respect to the current location of the swarm. To make an algorithm based on nest-site selection applicable to real optimization problems, the swarm

Algorithm 6 Bee Nest-Site Selection Scheme (BNSSS)

```
place bees on a random home position (swarm location) in the search space
initialize parameters $\Phi$, $\Phi'$, $\Psi$, $d$, and $d'$
while stopping criterion not satisfied do
    for all scouts do
        $k = h$
        repeat
            the scout flies to a random position $x$ with maximum distance $d$ to its current home
            if $f(x) \geq \Phi$ then
                the scout performs $k$ local search steps to find an improved location
                $k = 0$
            else
                $k = k - 1$
            end if
        until $k \leq= 0$
    end for
    for all followers do
        randomly assign the follower to one of the scouts where the probability depends on the quality of the location of the scout
        repeat
            the follower flies to a random position $x$ with maximum distance $d'$ from the location of the scout it is assigned to
            if $f(x) \geq \Phi'$ then
                the follower performs $k$ local search steps to find an improved location
                $k = 0$ and stops
            else
                $k = k - 1$
            end if
        until $k \leq= 0$
        if the follower could not find a location $x$ with $f(x) \geq \Phi'$ then
            the follower abandons the scout
        end if
    end for
    if the swarm has found a location that is better than it its home location then
        its new home is the best of these locations
    else if there exists a scout which has more than $\Psi$ followers assigned to it then
        the swarm is assigned to the scout or one of its followers which has the best location
    else
        the swarm is assigned a new randomly chosen home location or it stays at its current location
    end if
    update $d$, $d'$, $\Phi$, $\Phi'$ and $\Psi$
end while
```

needs to become more sensitive to small quality differences to identify better potential nest sites when the swarm comes closer to the location of an optimum.

The decision-making process underlying the optimization is slow. The higher the quorum q of bees needed at a potential nest before the swarm changes to this site, the slower is the optimization process. The quorum mechanism could however also prove to be useful in terms of optimization, because the existence of a quorum prevents premature convergence onto a local minimum by slowing down the decision-process and thus giving better sites a higher chance to be discovered and enter the decision-making process. Another potential benefit of the quorum mechanism is that it requires bees to revisit and reassess a given site several times, which is important for dynamic or noisy optimization functions.

8.3. The BNSSS scheme

Since some aspects of the bee nest site selection model are relevant for real bees but are not useful for a function optimization algorithm, we present here a scheme — called Bee Nest-Site Selection Scheme (BNSSS) — for the design of optimization algorithms. The BNSSS, described in Algorithm 6, is provided as a framework into which details have to be added when a specific algorithm is designed. For example the values of d, d', Φ, Φ', and Ψ have to be defined. Where Φ, Φ', and Ψ should depend on the quality of the locations that have been found already. The values for d and d' might decrease during the run of the algorithm so that the swarm concentrates on a small area of the search space. In contrast, for dynamic optimization functions it might also be necessary to increase the values of d and d' at points in the decision-making process when it is found that the function to be optimized has changed. In addition, for noisy optimization functions it might be suitable to set $d' = 0$ so that the location of a scout is evaluated several times.

The BNSSS scheme given here is designed for a single swarm of bees, however the application of multiple swarms within an algorithm is also possible. This would require defining how the different swarms cooperate – for example, the swarms might be implemented to remain a certain distance from each other in order to cover different parts of the search space.

8.4. Applying bee nest-site selection behaviour to molecular docking

As we have already seen, the nest-site selection process has great potential for optimization. Based on the BNSSS scheme outlined in the last section, the first Bee Nest-site optimization (BNSO) algorithm for solving a complex optimization problem was devel-

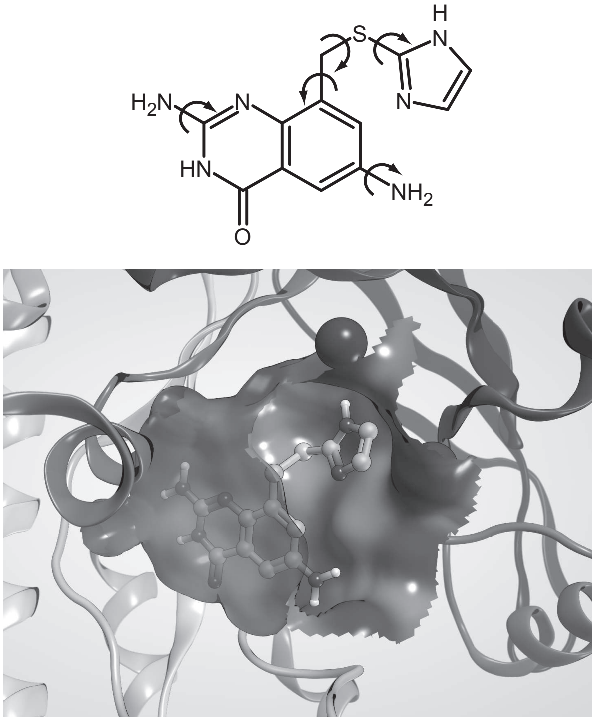


Figure 8.4.: A small molecular ligand visualised as a chemical formula with internal degrees of freedom highlighted by arrows (top) and the ligand bound into the pocket of a tRNA-Guanine Transglycosylase (bottom) as experimentally resolved by X-ray crystallography (Protein Database entry 1K4G). This figure is reproduced from Diwold et al. (2011c) and was created by Carsten Baldauf.

oped (Himmelbach, 2011). This algorithm is called Bee-Nest and its application in the domain of molecular docking, which constitutes a challenging real-life application domain, is outlined in this section. Unless stated otherwise the figures in this section (adapted from Diwold et al. (2011c)) were originally created by Daniel Himmelbach.

By recognizing small molecules, proteins act as the receptors of ligands. These interactions are formed if the three dimensional (3D) structure of the ligand fits into the binding pocket of the protein, like a key into a lock (see Figure 8.4 for an exemplary illustration). Knowledge about such interactions is crucial for the understanding of physiological processes and is a fundamental basis for the development of pharmaceutical substances.

The 3D structural information has been experimentally resolved for only a limited number of protein-ligand pairs, while no such data is available for the vast majority of cases. As resolving structural ligand-protein information experimentally is quite costly, computational approaches have become more and more established in the prediction of such complexes (Halperin et al., 2002). Computational approaches allow the fast and inexpensive screening of large libraries of potential ligands against a variety of protein targets and thus serve as a means of sampling potential ligand candidates, with the best results then being further investigated in wet-lab experiments. Such rapid *in silico*-screening methods are of growing importance in the industrial drug design process.

From a computer science perspective, molecular docking boils down to an optimization problem, namely finding the protein-ligand pose with minimal binding energy. Given a scoring function that estimates the binding energy of a protein-ligand complex, the docking problem results in the search for the global minimum in a multi-dimensional energy landscape.

Several population-based metaheuristics such as genetic algorithms (Jones et al., 1995), ant colony optimization (Korb et al., 2006) and particle swarm optimization (Janson et al., 2008; Liu et al., 2005; Meier et al., 2010) have been proposed to provide solutions to the problem of molecular docking.

8.4.1. Bee-Nest algorithm

The Bee-Nest optimization starts with a colony of virtual bees being placed at a random position in search space. Here the search space represents an environment and each position in the search space corresponds to a potential nest-site (solution). The quality of a nest-site is given by the value of the function to be optimized at the corresponding position.

Using the principles of nest-site selection the colony tries to find a nest-site of better quality than its current location. A colony contains two types of bees: *scouts* and *followers*. The selection process begins with scouts trying to find potential nest-sites in the surroundings of the swarm's current location. If the scouts are able to find a location that is of acceptable quality, they report it to the swarm. Followers choose a scout to follow based on the quality of the nest-site it has found (i.e., scouts that found better nest-sites will attract more followers). The follower then flies to the location the scout found and searches the surrounding to eventually find a better location. If the colony is able to come up with a location that is of better quality than its current location, it will relocate itself to the new location and restart the nest-site selection process. Otherwise, the colony repeats the selection process at its current location.

More formally: Given a dim dimensional function F that is to be minimized and a swarm of n virtual bees consisting of n_{scout} scouts and $n_{follower}$ followers (i.e., $n = n_{scout} + n_{follower}$). The swarm is initially placed on a randomly chosen location $p_{swarm} = (x_1, \ldots, x_{dim})$ in the search space. Each scout s chooses a location p_s uniformly at random with the restriction that it has a maximal distance $d_{scout} \cdot f_r$ to the swarm's current location (i.e., $|p_{swarm} - p_s| \leq d_{scout} \cdot f_r$). Here d_{scout} is a parameter and f_r $(0 \leq f_r \leq 1)$ is a factor that decreases over time in order to achieve an increasingly local search of the

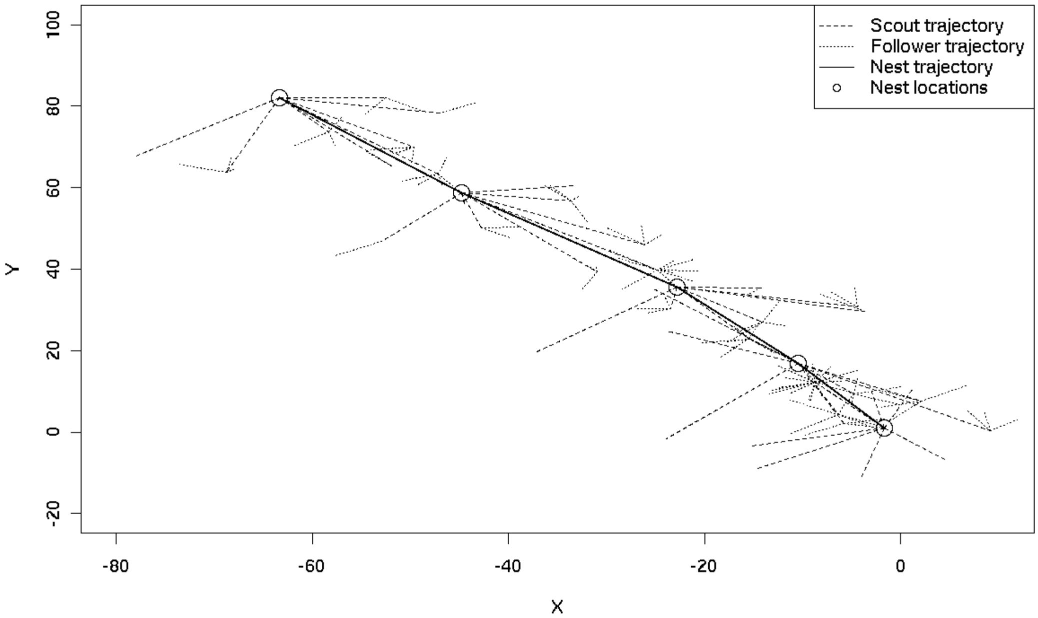


Figure 8.5.: Visualization of a Bee-Nest optimization run over 4 nest relocations (in this case on a two dimensional Sphere benchmark function).

algorithm. One possibility of defining f_r is to predefine a maximum number of iterations $MAXITER$ per optimization run and adapt f_r accordingly by

$$f_r = 1 - \frac{iteration}{MAXITER} \tag{8.7}$$

where $iteration$ is the number of the current iteration. The quality of the chosen location is tested with the criterion $F(p_s) \leq F(p_{swarm}) \cdot f_q$, where parameter f_q $(0 \leq f_q \leq 1)$ is a quality factor. If a chosen location satisfies the criterion, this means the scout has found a potential nest-site at location p_s, which can then be chosen by potential followers. The probability to be chosen by a follower depends on its relative fitness, defined by $fit_s = \max\{0, (F(p_{swarm}) \cdot f_q) - F(p_s)\}$.

After each scout has updated its location, each follower f chooses one scout using a standard roulette wheel selection so that the probability P_s of choosing scout s is

$$P_s = \frac{fit_s}{\sum_{k=1}^{n_{scout}} fit_k}. \tag{8.8}$$

Each follower is placed at the location of the scout it has chosen and chooses uniformly at random a location p_f in the vicinity of the scout's location p_s, such that it is not further

Algorithm 7 Bee-Nest

```
place swarm on random location $p$, i.e., $p_{swarm} = p$
repeats = 0;
while stop criterion not satisfied do
    reduce $f_{range}$ according to Eq. 8.7
    for all scouts do
        Choose new location $p_s$ with a max distance of $d_{scout} \cdot f_{range}$ to the nest
        $fit_s = \max\{0, (F(p_{swarm}) \cdot f_q) - F(p_s)\}$
    end for
    for all followers do
        Choose a scout $s$ according to Eq. 8.8
        Choose new location $p_{follower}$ with a max distance of $d_{follower}$ to chosen scout's
        position $p_s$
        Sample search space between $p_s$ and $p_{follower}$ in $m$ flight steps
    end for
    if better location $p$ was found then
        Relocate swarm to $p$, i.e., $p_{swarm} = p$
    else
        if $repeats \geq MAXREPEATS$ then
            Place swarm on new random location $p$, i.e., $p_{swarm} = p$
            repeats=0;
        else
            repeats=repeats+1;
        end if
    end if
end while
```

distant than the parameter $d_{follower}$ (i.e., $|p_s - p_f| \leq d_{follower}$). Then the follower samples the search space between p_s and p_f in a directed flight consisting of m equal length flight steps. At the end of each step function F is evaluated.

During the whole process the system maintains the best solution found so far p_{best}. If the swarm is able to find a better location than its current location (i.e., $F(p_{best}) > F(p_{swarm})$) it migrates to the new location. Otherwise it restarts the nest-site selection process from its current location. If a swarm is not able to improve its location in $MAXREPEATS$ steps then it is moved to a random location in the search space and the nest-site selection process restarts. The algorithm terminates when a given stopping criterion is satisfied.

For better understanding, the pseudocode of the Bee-Nest algorithm is presented in Algorithm 7. A visualization of a search run over 4 nest-site relocations, containing the search trajectories of scouts and followers, on a 2-dimensional Sphere function is shown in Figure 8.5.

8.4.2. Experimental setup

Bee-Nest was implemented in ParaDocks, a molecular docking framework developed for population-based heuristics (Meier et al., 2010). To benchmark the performance of the Bee-Nest algorithm for the docking problem, 173 instances from the PDBbind database *core-set* (Wang et al., 2004) were used for testing. The obtained optimization and sampling performance was compared to a PSO algorithm that was previously proposed for docking (Meier et al., 2010), as well as randomly selected solutions and solutions derived using local optimization.

Modelling of molecules and molecular complexes in chemistry and biochemistry always features a variety of approximations, some of which affect the number of degrees of freedom (DOF), i.e., the dimension of the search space. In the approximation used here, a ligand-receptor pose is described by a vector containing 3 Cartesian coordinates for the ligand's position, its orientation is described by the 4 DOF of a quaternion, and N internal DOF describe the ligand's conformation. In the used test instances the internal flexibility ranges up to $N = 35$ internal DOF. Thus, the search space with $7 + N$ dimensions can be up to 42-dimensional. The conformation of the receptor is regarded to be rigid (this is an accepted approximation in the field).

The statistically derived potential PMF04 was used to describe the binding energy landscape between ligand and receptor as pair-wise potentials of ligand and receptor atoms:

$$W_{ij}(d_{ij}) = -\ln \frac{g_{ij}(d_{ij})}{g_{ref}}, \tag{8.9}$$

with $g_{ij}(d_{ij})$ the density of the atom pair ij in distance d_{ij}, and g_{ref} the average density of atom pair ij. PMF04 is derived from 6611 protein ligand complexes and describes the interactions of 17 protein atom types with 34 ligand atom types (for a detailed description, see Muegge 2006). The adaptations necessary to use PMF04 for molecular docking are described in Meier et al. (2010).

The following three optimization algorithms were employed as a reference:

PSO: The PSO was used with the settings suggested in Meier et al. (2010) with 30 particles evaluated in 300,000 generations.

RNDM: Nine million random poses were generated based on the Mersenne twister algorithm published by Matsumoto and Nishimura (1998) and the best result was kept.

RHC: 9,000 randomly chosen poses were locally optimized by 1,000 hill climbing steps. Lower energy poses are accepted, higher energy poses are discarded.

Algorithm 8 Random Walk

```
for $k \in 0 \ldots MaxLO$ do
    $f_l = (1 - k/MaxLO)/16$
    Generate new random solution $p_r$ with $|p_r - p_{nn}| < f_l \cdot d_{scout}$
    if $(F(p_r) < F(p_{nn})$ then
        $p_{nn} = p_r$
    end if
end for
```

For the molecular docking problem the BNSO algorithm Bee-Nest was slightly extended with a local search as follows. When a better nest-site p_{nn} was found, by a scout or follower, a simple random walk (Algorithm 8), was applied to the location $MaxLO \geq 1$ times for the purpose of local optimization. This random walk generator chooses a location in the vicinity of the current best location p_{nn}. The maximum distance of the randomly generated location p_r to the current best location p_{nn} is restricted to $|p_r - p_{nn}| < f_l \cdot d_{scout}$ where f_l is a parameter. Parameter f_l decreases over the the local search iterations towards 0 (see Algorithm 8), which leads to convergence of the new locations p_r to p_{nn}. The random walk is also applied to the final location returned by Bee-Nest for $PostLO$ times.

The following parameter settings were used for the BeeNest algorithm: $n = 30$, $n_{scout} = 10$, $n_{follower} = 20$, $f_q = 0.95$, $MAXREPEATS = 20$, $MaxLO = 4$, $PostLO = 4096$. Since in the context of molecular docking the dimensions of the search space correspond to different aspects of the problem (position, orientation, rotations of single axes in the molecules (internal DOF)) different values of d_{scout} ($d_{follower}$) are used for the different types of dimensions in order to determine the search range for new locations around the current nest or scout location:

$$d_{scout} = \begin{cases} 0.003616 \cdot \mathit{SpaceRange}, & \text{for position} \\ 0.001084 \cdot 2\pi, & \text{for orientation} \\ 0.027854 \cdot 2\pi, & \text{for internal DOFs} \end{cases}$$

$$d_{follower} = \begin{cases} 0.025366 \cdot \mathit{SpaceRange}, & \text{for position,} \\ 0.039257 \cdot 2\pi, & \text{for orientation} \\ 0.012289 \cdot 2\pi, & \text{for internal DOFs} \end{cases}$$

Each of the four algorithms were tested on the 173 test instances, with a duration corresponding to $9,000,000$ energy evaluations. Each test instance was repeated 50 times. The test results will provide on insight into the quality of the solutions and the robustness of the algorithms with regards to the molecular docking problem.

8.4.3. Results

Tables 8.3, 8.4 and 8.5 show a comparison of the minimum, median and mean energy values achieved by the four algorithms averaged over all test-instances. As can be seen, Bee-Nest performs very well. It is able to achieve better energy values than PSO as well as RHC and RNDM on the majority of the test instances for all three comparisons. The random hill climbing method (RHC) shows a decent performance, which is slightly worse than PSO and Bee-Nest. The randomly generated solutions of RNDM are outperformed in each aspect by the other algorithms. Table 8.3 suggests that Bee-Nest is particularly capable of finding very low energy levels. In comparison with PSO it found the protein conformations with the lowest energy levels in 141 of the 173 test instances.

Table 8.3.: Minimum energy value comparison for 173 test instances. Each cell denotes the number of test instances for which the minimum energy value obtained in 50 test runs of the algorithm stated in the row was better (i.e., lower) than the minimum energy of the algorithm stated in the column.

Alg vs. Alg	PSO	Bee-Nest	RNDM	RHC
PSO	-	32	172	134
Bee-Nest	141	-	173	168
RNDM	1	0	-	0
RHC	39	5	173	-

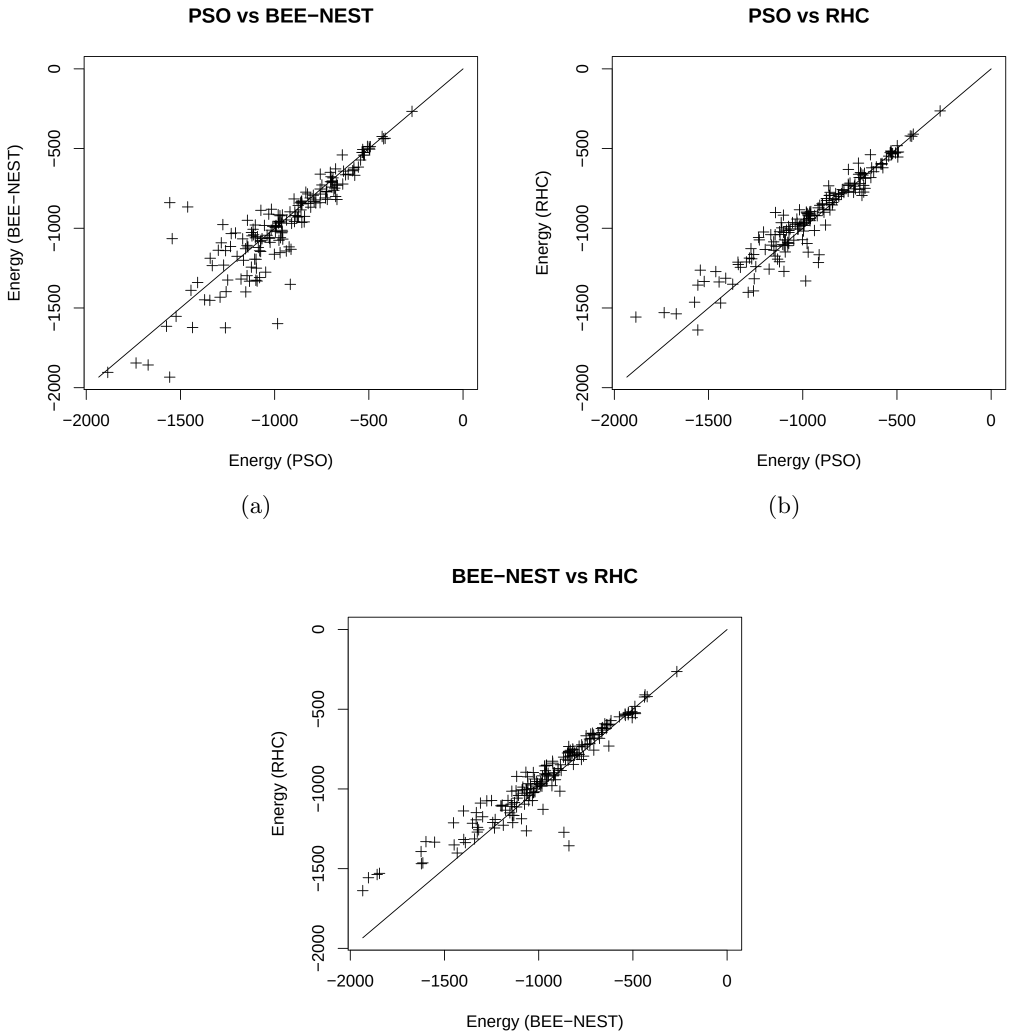
Table 8.4.: Median energy value comparison (analogously to Table 8.3).

Alg vs. Alg	PSO	Bee-Nest	RNDM	RHC
PSO	-	74	173	116
Bee-Nest	99	-	171	142
RNDM	1	0	-	0
RHC	39	5	173	-

Table 8.5.: Mean energy value comparison (analogously to Table 8.3).

Alg vs. Alg	PSO	Bee-Nest	RNDM	RHC
PSO	-	77	173	113
Bee-Nest	96	-	173	133
RNDM	0	0	-	0
RHC	60	40	173	-

Figure 8.6 depicts scatter plots of the median energy levels found by the Bee-Nest, PSO, and RHC in all test instances. Scatter plots for RNDM are omitted as its performance was very poor in general (see Tables 8.3-8.5). In Figure 8.6 the x-value corresponds to the median energy value of one algorithm for a given molecular docking instance and the



(a)

(b)

(c)

Figure 8.6.: Scatter plots comparing the median performance (energy) of two algorithms for all test instances. Each data point indicates the contrasted algorithms' performances on a specific test instance. Points lying on the diagonal reflect comparable performance by each algorithm. As lower energy reflects better performance, points above the diagonal indicate better performance by the algorithm indicated on the x-axis, and vice versa.

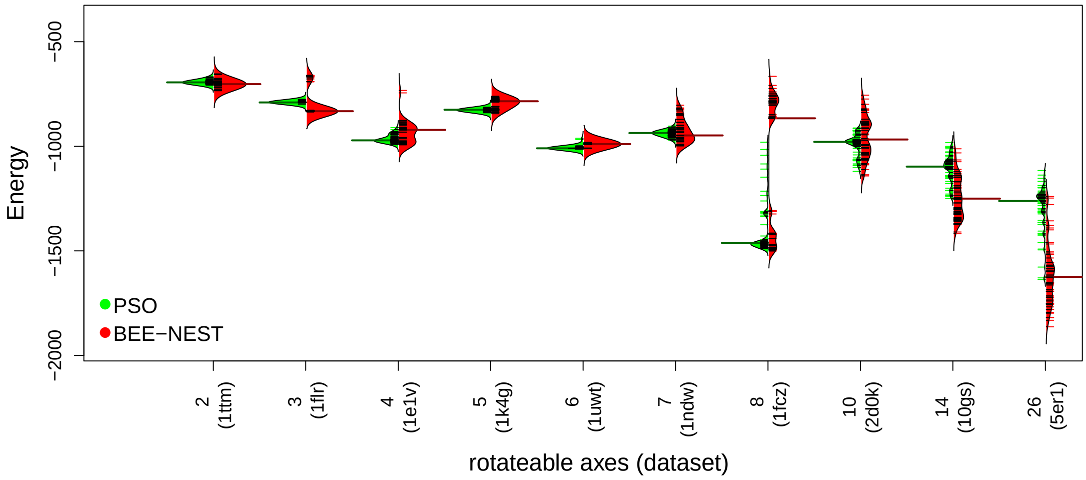


Figure 8.7.: Bean plot of representative test instances from the test set. Each bean depicts the performance (energy) distribution of the PSO (green) and Bee-Nest (red) arising from 50 repeats of each instance (x-axis). Instances are ordered by number of rotatable axes. Small coloured lines depict individual data points; dark lines show distribution mean.

respective y-value corresponds to the energy value of a reference algorithm for the same docking instance. Values on or close to the diagonal denote test-cases where the algorithms showed similar performance. Values above the diagonal correspond to instances where the algorithm on the x-axis achieved better energy values and values below denote instances where the algorithm on the y-axis produced better energy values.

As can be seen in Figure 8.6(a) PSO and Bee-Nest perform on par in instances with high energy levels (which usually corresponds to proteins with a small number of rotatable axes). In comparison to PSO the performance of Bee-Nest improves for instances with a higher number of rotatable axes in the ligand. This can also be observed when comparing the Bee-Nest with the RHC.

Figure 8.7 shows beanplots (see Kampstra (2008) for more details) depicting the estimated energy level distributions of the 50 solutions found by Bee-Nest and PSO for a representative subset of docking instances from the test set. As can be seen, the spread and thus the solution diversity increases with the increase of the internal flexibility of the ligand (number of rotatable axes). This is not surprising as it directly increases the dimensionality of the search space and thus leads to a more complex fitness landscape. In cases of an increased number of rotatable axes, the distribution of the PSO's energy levels is quite narrow in comparison to Bee-Nest's energy level distribution. This suggests that PSO generates protein ligand poses that are similar. In comparison Bee-Nest is more

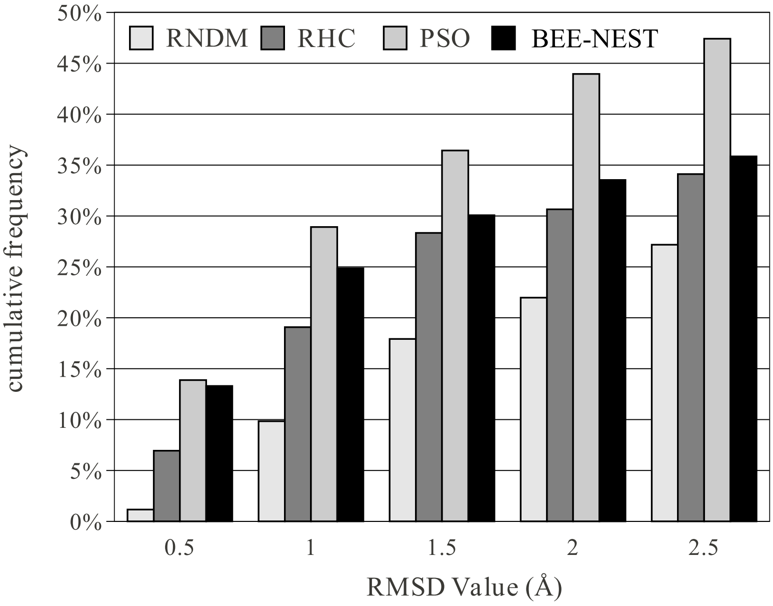


Figure 8.8.: Cumulative histogram of the RMSD values from the X-ray crystal structures of the best solutions. This figure is adapted from DIWOLD ET AL. (2011c) and was originally created by Carsten Baldauf.

likely to produce a variety of poses during the 50 test runs, especially for proteins with many rotatable axes. Furthermore, the behaviour illustrated in the plots (Figure 8.7) also suggests that Bee-Nest has some problems escaping local optima. This is best seen for the docking instance $1fcz$, where the Bee-Nest converges to either one of two possible minima, one with a suboptimal energy level around -800 and one with an optimal energy level around -1300. In contrast, PSO converges towards a configuration at the low energy level in most of the cases. However, such a spread can also be beneficial; for example in the docking instances $10gs$ and $5er1$, Bee-Nest is able to reach lower energy levels, whereas the PSO is apparently stuck in suboptimal configurations.

Root mean square deviation As pointed out above, the energy levels of the protein configurations found by Bee-Nest in comparison to the reference algorithms are promising as they are in general of lower energy. In order to judge the biological significance of the calculated poses, the root mean square deviation (RMSD) was calculated for the best results found for each instance by each algorithm. RMSD is often used to measures the average distance between the different conformations of molecules. Here we compute the RMSD of the poses generated by our docking experiments with respect to experimentally derived 3D structures resolved by X-ray crystallography. Thus, the RMSD value is a good estimate for the biological plausibility of the calculated conformation. RMSD gives the deviation of the algorithmically generated protein-ligand pose from the experimentally generated reference in Ångström ($0.1nm$). As proteins are not rigid bodies in space, RMSD values of up to $2.5Å$ can be considered as a reasonable fit.

Figure 8.8 depicts a cumulative histogram of the RMSD values of the solutions generated by the four algorithms RNDM, RHC, PSO, and Bee-Nest. The solutions produced by the PSO show the best fit regarding the real position and conformation of the ligand in the receptor. While Bee-Nest and PSO produce roughly the same amount of conformations that are a very close fit (i.e., 13% and 14% of the poses produced by the Bee-Nest and PSO, respectively, have an RMSD $\leq 0.5Å$), this does not hold for higher RMSD values. Only 36% of the solutions found by Bee-Nest have a RMSD value $\leq 2.5Å$, whereas this is the case for 47% of the conformations produced by PSO. This result is unexpected, as it was shown in the last section that the energy levels of the conformations produced by Bee-Nest are in general lower than those of PSO conformations.

There are two potential explanations for this observation: As outlined earlier, receptor-ligand conformations are evaluated using approximate energy functions to estimate their energy. Thus, part of the problem can come from the accuracy of the scoring function. It could, for example, be the case that the low-energy conformations found by Bee-Nest are not plausible in comparison to the real conformation. However, this can only explain a part of this odd behaviour, as this argument also applies to the solutions generated by the PSO. Another explanation for this phenomenon is that even though Bee-Nest sometimes gets stuck in local optima, it is still able to adapt the conformation of the ligand in such a way that it leads to low energy values. This would highlight the ability of Bee-Nest to generate low-energy solutions, but also shows its limited ability in overcoming larger energy barriers during the optimization process, as it has a single position (i.e., receptor-ligand pose) as a starting point which is then continuously improved. In contrast, PSO performs a more thorough global search, as it starts off with its particles distributed over the whole search space.

Molecular docking fitness landscapes are by no means a homogeneous environment. Usually, only a very limited number of conformations yield low energy levels and seemingly small variations in the conformations can lead to a drastic quality change. It could thus be the case that while Bee-Nest outperforms PSO in terms of fine-tuning the conformation of the protein-complex, it can not creep over the fitness barriers imposed by the fitness landscape as well as PSO. Both explanations will be further investigated in future work, for example by using different scoring functions. If the latter explanation turns out to be true, a hybrid approach in which PSO is used to sample the search space and Bee-Nest acts more as a fine-tuning mechanism, might yield an algorithm of truly improved performance.

8.5. Conclusions

Bee-inspired optimization techniques have recently become popular within the optimization community but have so far been restricted to using the bees' foraging behaviour and mating behaviour. This chapter proposed the bees' nest-site selection behaviour as a third class of behaviour to be used for the development of bee-inspired optimization techniques. Nest-site selection involves the active discovery of potential sites by scouting bees and a subsequent decision between those candidate sites. In nature, it enables bees to solve the best-of-n-problem (i.e., deciding on the best nest-site). In contrast to the mating and foraging behaviour, nest-site selection constitutes a decision-making process that has a clear optimum.

To investigate the optimization potential of the nest-site selection mechanism, a biological model was used. This model was initially introduced in Chapter 6, where it was used to study the extent to which the distribution of nest-sites within an environment shape the choices of a swarm. Three optimization experiments were conducted. Using this model, we performed three optimization experiments.

In the first experiment the swarm was situated in an environment where the quality of nest-sites could fluctuate over time. When presenting the swarm with two nesting options, where one was of stable but mediocre quality and the other was of unstable good quality, the swarm chose the more reliable stable nest-site. In the second experiment the swarm was placed in a dynamic environment where nest-sites swapped their qualities at regular intervals. Given such a scenario the swarm was able to re-decide on the better nest-site after such a quality change occurred. In the last experiment the swarm was placed on two-dimensional fitness landscapes. Each position in these landscapes represented a potential nest-site, with the quality of a site corresponding to the fitness function value of the respective position. Given an iterative application of the nest-site selection process, the biological nest-site selection model was able to achieve function optimization up to a certain extent.

These results corroborate that the honeybee's nest-site selection process is indeed useful in the context of optimization. However, the biological model can not directly be used for general function optimization problems as the observed optimization is coarse and slow due to its biologically realistic nature. We thus introduced a general algorithmic scheme called "Bee Nest-Site Selection Scheme" (BNSSS), which is inspired by the nest-site selection model and can be used in the design of optimization algorithms.

Based on the introduced optimization scheme the first bee nest-site optimization algorithm (BNSO algorithm) "Bee-Nest" was developed for the domain of molecular docking. The performance of Bee-Nest was compared to three reference algorithms that have been previously used in this problem domain.

Molecular docking was chosen as a test problem as it constitutes a challenging real-life optimization problem of high importance in the fields of bioinformatics and biochemistry. Bee-Nest was tested on an available set of molecular docking instances. The solutions found by Bee-Nest were compared to those established by the reference algorithms in terms of lowest energies, energy distribution and RMSD to the reference solution. In comparison to the three reference algorithms Bee-Nest is able to generate receptor-ligand conformations with the lowest energy levels for the majority of the test instances. However, the correspondence to empirical data of the conformations produced by Bee-Nest are not as accurate as their energy levels would suggest. The reason for this could be the used scoring function. Another potential explanation is that Bee-Nest has problems overcoming the vast fitness barriers imposed by the molecular docking fitness landscapes. Lower energy values would then be the result of the Bee-Nest's superiority in fine tuning the protein conformations regarding its surrounding. If this is the case, a hybrid approach where the PSO is applied as a means of search space sampling and the Bee-Nest algorithm functions as a post-processing algorithm might yield a very good performance if applied to molecular docking.

9. Conclusions

This work has investigated several self-organizing principles found in social insects which can be regarded as “optimization in nature”, as they enable a group of relatively simple homogeneous individuals to tackle complex tasks within their environment. Understanding such principles is not only in the interest of biology, but also yields a better understanding of the complex system behaviour underlying such self-organizational principles. Knowledge of such systems can be used in the design of new adaptive and robust control and optimization mechanisms, as well as to aid understanding of existing methods based on self-organizing principles found in nature. In addition to the investigation of self-organization in social insects, a new optimization technique was introduced which is based on the nest-site selection behaviour of honeybees. The optimization potential of the nest-site selection process was first evaluated in its biological context and an abstracted optimization scheme was presented. The first optimization algorithm based on this optimization scheme was outlined and tested on a difficult real-world problem.

9.1. Division of labour

The first part of this thesis investigated several aspects of division of labour in social insects on the basis of response threshold models. First the adaptiveness of different-sized colonies to dynamic changes in the environment was analysed. Our findings show that a colony’s ability to react fast to changes in the environment increases as a function of colony-size. As a determining factor for this colony-size dependent performance difference, the colony-size dependency of stimulus growth was identified, which directly affects an individual’s perception of a task’s necessity. In order to decrease the level of performance differences, different-sized colonies need to exhibit different flexibility. While small colonies need to be able to adapt fast to environmental changes, larger colonies can afford a more gradual adaptation on an individual level. If an individual’s adaptation speed is adjusted to the size of its colony, this will impact the specialisation present in different sized colonies. In a test of how well different-sized colonies can sustain task memory over consecutive task demand changes, we were able to show that large colonies are able to maintain specialists for a task over a long period of time, even when the task is of low demand, while small colonies “forget” about the task as soon as it is done. Colony-size dependent specialisation is a phenomenon often observed in nature, and while it has been directly linked to colony-size as well as competition before, our results suggest that colony-size dependent specialization is a direct consequence of the dynamic environment a social insect colony is embedded in. While small size colonies are more dependent on the work of each and every member and thus require very flexible individuals, larger colonies allow less flexibility which will lead to specialisation.

The second aspect of division of labour which was investigated is to what extent different spatial distributions of tasks and individuals influence division of labour. To study this, an extension of the threshold reinforcement model – the "spatial threshold reinforcement model" – was introduced, which enables the study of spatial task distribution across multiple locations in these models. Using this model, we were able to show that social insects can benefit from a spatial task separation within their environment, as it increases the colony's productivity. This could explain the spatial organization which is often observed within social insect colonies. As the maintenance of order does not come without cost, we also investigated to what extent it pays off to actively enforce order within a colony. Our findings suggest that even when the maintenance of spatial separation is associated with a cost (omission of workforce), it is still beneficial for a colony to actively maintain order within its environment. Additionally, an exploration of different sorting strategies showed that sorting mechanisms operating in parallel to the threshold model show little flexibility to adapt to changes in sorting demand. The best approach was to implement sorting as an additional task in the system, as this allows the colony to best adapt its sorting-force to the required demand.

Our results on division of labour on the basis of response threshold models corroborate empirical observations and deliver potential explanations for them. However, as several control mechanisms proposed in the context of scheduling and optimization are based on the principles of division of labour in social insects, these findings might also help to further analyse and understand the performance of such systems as well as aid to further improve them and increase their applicability to other domains.

9.2. Self-organization in honeybees

In the second part of this thesis, several aspects of self-organization found in honeybees were studied using computer-simulations. First we showed the influence of spatial nest-site distribution on the ability of the European honeybee *A. mellifera* to select a new nest-site. When a swarm is situated in an environment that supplies a dense congregation of potential new nest-sites, this can impact the selection process and prevent convergence to a single site. This is due to the fact that honeybees that miss a potential nest-site they were recruited for are likely to discover an alternative which then will enter the nest-site selection process as a competitor. As such "missing" behaviour is triggered by the amount of dances a bee follows, nest-site quality as well as angular distribution between the nest-sites were identified as driving factors of this behaviour. When a swarm is confronted with two regions of differing nest-site density, the swarm will be biased towards the region where the sites are more densely clustered. While a bias towards nest-site rich habitats can be observed if all the nest-sites present in the selection process are of equal quality, a swarm's ability to choose the best nest-site in an environment is not obstructed and it will choose a nest-site in the sparser environment if this nest-site is of superior quality. A colony's disability to disregard close-by nest-sites which might prevent convergence to a single nest-site might explain why honeybee species which have less requirements regarding a new nest-site have evolved a more imprecise form of nest-site selection than cavity-nesting species, as too much precision might hinder a swarm from its decision.

A second self-organizing behaviour of honeybee which was investigated is the group guidance behaviour of migrating honeybee swarms. Previous empirical studies have suggested that migrating swarms are guided actively by fast-flying informed individuals (streakers) that consecutively fly through the upper segment of a swarm. Non-streaking individuals are thought to chase after faster individuals which leads to a directed motion of the swarm. An alternative hypothesis, which could not be ruled out until now, is that migrating honeybee swarms are guided passively. Passive guidance assumes that informed individuals only differ from uninformed ones in a slight directional bias, and in a moving group this directional bias will drag the swarm in the desired direction. To investigate which of these two rivalling hypotheses better captures honeybee swarm flight characteristics, a movement model was used which can exhibit both forms of guidance and allows their study and comparison under equal parameter conditions.

Our results suggest that active guidance, as proposed by previous empirical research, better reflects moving honeybee swarms than passive guidance. Unlike other moving animal collectives such as fish or locusts, where individuals are in general well aligned in terms of directionality and speed, honeybee swarms have been found to be very noisy in terms of individual flight speed and directional orientation. This noise is not reproducible under passive guidance as the indistinguishability between informed and uninformed individuals leads to the incorporation of a lot of information in an individual's update on directionality. Active guidance in contrast allows such a distinction and is thus able to replicate characteristics of real swarms. These results thus corroborate active guidance as the mechanism underlying migration in honeybee. In addition they emphasise that the idiosyncrasy underlying the individual behaviour in animal collectives has evolved according to the needs of the respective species and thus can take different forms.

As our results suggest that active guidance underlies the migration behaviour of honeybee swarms, we also tested to what extent this guidance mechanism is able to incorporate directional dissent among informed individuals. The consensus on a nest-site before lift-off can vary across different honeybee species, and active guidance must be able to deal with such directional dissent. We were able to demonstrate that group movement can also be achieved using active guidance under directional dissent. In such situations the travel direction of the swarm represents the average direction of the informed individuals, which can be interpreted as an in-flight decision-making process.

Nest-site selection in honeybees can be regarded as an natural optimization process. It is based on simple rules and achieves local optimization as it enables a swarm to decide between several potential nest-sites in an previously unknown dynamic environment. These factors make the nest-site selection process interesting in the context of function optimization. To investigate this optimization potential, the aforementioned spatial nest-site selection model was used. We first tested a swarm's ability to choose a nest-site in a stochastic environment where nest-sites can exhibit fluctuations in quality. In such a situation the swarm is able to select a stable site and disregard unstable solutions, although unstable solutions transiently have better quality than the stable solution. In a second experiment we tested the swarm's ability to adapt its decision making in a dynamic environment where nest-site qualities are frequently swapped. Again, the nest-site selection process was shown to enable optimization, as the swarm could track the nest-site with the highest quality. In a final experiment the swarm was placed on 2-dimensional fitness landscapes, where each position represented a potential nest-site. Over an iterative application of the nest-site selection process we were able to achieve function optimization up to a certain extent.

9.3. A nest-site inspired optimization framework

Based on these findings, which suggest that the nest-site selection process is indeed useful in the context of optimization, we introduced a general algorithmic scheme — called "Bee Nest-Site Selection Scheme" (BNSSS) — which is inspired by the nest-site selection model and can be used in the design of optimization algorithms. In addition, we developed the first bee nest-site optimization algorithm "Bee-Nest". Bee-Nest was applied to the domain of protein docking. The performance results were encouraging in comparison to other optimization techniques for this problem domain, as Bee-Nest was able to achieve very good results in terms of creating docking configurations with minimal binding energy. However, a limitation is that some of the Bee-Nest solutions do not correspond to empirical results. It is unclear whether this is due to the scoring function used, or if it reflects Bee-Nest's inability to overcome fitness barriers in the search space. Nevertheless, Bee-Nest shows potential for future applications due to its strength in local search, and could perhaps be used as part of a hybrid optimization system.

9.4. Closing remarks

In conclusion this thesis has brought forward several models which allow the study of self-organization in social insects. Using these models a range of issues that are of biological interest were addressed. In addition, we were able to demonstrate that there are still many unexplored biological mechanisms which may be of interest for computer scientists working on computational solutions to optimization problems.

Bibliography

Abbass HA (2001a). Marriage in honeybees optimization (MBO): A haplometrosis polygynous swarming approach. In *Proceedings of the Congress on Evolutionary Computation (CEC '01)*, volume 1, 207–214. IEEE Computer Society.

Abbass HA (2001b). A single queen single worker honey bees approach to 3-sat. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '01)*, 807–814. Morgan Kaufmann.

Afshar A, Haddad OB, Marino M, and Adams B (2007). Honey-bee mating optimization (HBMO) algorithm for optimal reservoir operation. In *Journal of the Franklin Institute*, 344(5): 452–462.

Akay B and Karaboga D (2009). Parameter Tuning for the Artificial Bee Colony Algorithm. In N Nguyen, R Kowalczyk, and SM Chen (editors), *Computational Collective Intelligence. Semantic Web, Social Networks and Multiagent Systems*, volume 5796 of *Lecture Notes in Computer Science*, 608–619. Springer Berlin / Heidelberg.

Amdam GV and Omholt SW (2003). The hive bee to forager transition in honeybee colonies: the double repressor hypothesis. In *Journal of Theoretical Biology*, 223(4): 451–464.

Anderson JA and Rosenfeld E (editors) (1998). *Talking nets : an oral history of neural networks*. MIT Press, Cambridge, MA, USA.

Avitabile A, Morse RA, and Boch R (1975). Swarming Honey Bees Guided by Pheromones. In *Annals of the Entomological Society of America*, 68(6): 1079–1082.

Bahamish HAA, Abdullah R, and Salam RA (2009). Protein Tertiary Structure Prediction Using Artificial Bee Colony Algorithm. In *Proceedings of the 3rd Asia International Conference on Modelling and Simulation (AMS '09)*, volume 0, 258–263. IEEE Computer Society.

Ballerini M, Cabibbo N, Candelier R, Cavagna A, Cisbani E, Giardina I, Lecomte V, Orlandi A, Parisi G, Procaccini A, Viale M, and Zdravkovic V (2008). Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study. In *Proceedings of the National Academy of Sciences*, 105(4): 1232–1237.

Bartumeus F, Luz MGED, Viswanathan GM, and Catalan J (2005). Animal search strategies: a quantitative random–walk analysis. In *Ecology*, 86(11): 3078–3087.

Baschelet E (1981). *Circular Statistics in biology.* Academic Press, New York, New York, NY, USA.

Baum KA, Rubink WL, Pinto MA, and Coulson RN (2005). Spatial and Temporal Distribution and Nest Site Characteristics of Feral Honey Bee (Hymenoptera: Apidae) Colonies in a Costal Prairie Landscape. In *Environmental Entomology*, 34(3): 610–618.

Bazazi S, Buhl J, Hale JJ, Anstey ML, Sword GA, Simpson SJ, and Couzin ID (2008). Collective Motion and Cannibalism in Locust Migratory Bands. In *Current Biology*, 18(10): 735–739.

Beckers R, Deneubourg JL, and Goss S (1993). Modulation of trail laying in the ant *Lasius niger* (Hymenoptera, Formicidae) and its role in the collective selection of a food source. In *Journal of Insect Behavior*, 6(6): 751–759.

Beckers R, Deneubourg JL, Goss S, and Pasteels JM (1990). Collective decision making through food recruitement. In *Insectes Sociaux*, 37(3): 258–267.

Beekman M (2005). How long will honey bees (*Apis mellifera* L.) be stimulated by scent to revisit past-profitable forage sites? In *Journal of Comparative Physiology A*, 191(12): 1115–1120.

Beekman M, Fathke RL, and Seeley TD (2006). How does an informed minority of scouts guide a honeybee swarm as it flies to its new home? In *Animal Behaviour*, 71(1): 161–171.

Beekman M, Gilchrist AL, Duncan M, and Sumpter DJT (2007). What makes a honeybee scout? In *Behavioral Ecology and Sociobiology*, 61(7): 985–995.

Beekman M, Gloag RS, Even N, Wattanachaiyingcharoen W, and Oldroyd BP (2008). Dance precision of *Apis florea* - clues to the evolution of the honeybee dance language? In *Behavioral Ecology and Sociobiology*, 62(8): 1259–1265.

Beekman M and Lew JB (2008). Foraging in honeybees – when does it pay to dance? In *Behavioral Ecology*, 19(2): 255–261.

Beekman M and Ratnieks FLW (2000). Long range foraging by the honeybee *Apis mellifera* L. In *Functional Ecology*, 14(4): 490–496.

Beekman M, Sumpter DJT, Seraphides N, and Ratnieks FLW (2004). Comparing foraging behaviour of small and large honey bee colonies by decoding waggle dances made by foragers. In *Functional Ecology*, 18(6): 829–835.

Bell WJ (1990). Searching behavior patterns in insects. In *Annual Reviews of Entomology*, 35: 447–467.

Ben-Shahar Y, Thompson CK, Hartz SM, Smith BH, and Robinson GE (2000). Differences in performance on a reversal learning test and division of labor in honey bee colonies. In *Animinal Cognition*, 3(3): 119–125.

Benhamou S (2007). How many animals really do the lévy walk? In *Ecology*, 88(8): 1962–1969.

Beni G and Wang J (1989). Swarm intelligence in cellular robotic systems. In *Proceedings of the NATO Advanced Workshop Robots and Biological Systems.* Tuscany, Italy.

Bénichou O, Coppey M, Moreau M, Suet PH, and Voituriez R (2005). Optimal Search Strategies for Hidden Targets. In *Physical Review Letters*, 94(19): 198101.

Berens P (2009). CircStat: A MATLAB Toolbox for Circular Statistics. In *Journal of Statistical Software*, 31(10): 1–21.

Bergman CM, Schaefer JA, and Luttich SN (2000). Caribou movement as correlated random walk. In *Oecologia*, 123(3): 364–374.

Bernardino A, Bernardino E, Sánchez-Pérez J, Gómez-Pulido J, and Vega-Rodríguez M (2011). Efficient Load Balancing Using the Bees Algorithm. In K Mehrotra, C Mohan, J Oh, P Varshney, and M Ali (editors), *Modern Approaches in Applied Intelligence*, volume 6704 of *Lecture Notes in Computer Science*, 469–479. Springer Berlin / Heidelberg.

Berthold P, Helbig AJ, Mohr G, and Querner U (1992). Rapid microevolution of migratory behaviour in a wild bird species. In *Nature*, 360(6405): 668–670.

Berthold P and Querner U (1981). Genetic basis of migratory behaviour in European warblers. In *Science*, 212(4490): 77–79.

Beshers SN, Huang ZY, Oono Y, and Robinson GE (2001). Social Inhibition and the Regulation of Temporal Polyethism in Honey Bees. In *Journal of Theoretical Biology*, 213(3): 461–479.

Boinski S and Garber PA (editors) (2000). *On the move: how and why animals travel in groups.* The University of Chicago Press, Chicago, IL, USA.

Bonabeau E, Dorigo M, and Theraulaz G (1999). *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, New York, NY, USA.

Bonabeau E, Sobkowski A, Theraulaz G, and Deneubourg JL (1997). Adaptive Task Allocation Inspired by a Model of Division of Labor in Social Insects. In *Biocomputing and emergent computation: Proceedings of BCEC97*, 36–45. World Scientific Press.

Bonabeau E, Theraulaz G, and Deneubourg JL (1996). Quantitative study of the fixed threshold model for the regulation of division of labour in insect societies. In *Proceedings of the Royal Society. Series B: Biological Sciences*, 263(1376): 1565–1569.

Bonabeau E, Theraulaz G, and Deneubourg JL (1998). Fixed response thresholds and the regulation of division of labor in insect societies. In *Bulletin of Mathematical Biology*, 60(4): 753–807.

Britton NF, Franks NR, Pratt SC, and Seeley TD (2002). Deciding on a new home: how do honeybees agree? In *Proceedings of the Royal Society. Series B: Biological Sciences*, 269(1498): 1383–1388.

Calderone NW (1998). Proximate mechanisms of age polyethism in the honey bee, *Apis mellifera* L. In *Apidologie*, 29(1-2): 127–158.

Camazine S, Visscher PK, Finley J, and Vetter RS (1999). House-hunting by honey bee swarms: collective decisions and individual behaviors. In *Insectes Sociaux*, 46(4): 348–360.

Campos M, Bonabeau E, Theraulaz G, and Deneubourg JL (2000). Dynamic Scheduling and Division of Labor in Social Insects. In *Adaptive Behavior*, 8(2): 83–95.

Chong CS, Low MYH, Sivakumar AI, and Gay KL (2006). A bee colony optimization algorithm to job shop scheduling. In *Proceedings of the 38th conference on Winter simulation (WSC '06)*, 1954–1961. ACM.

Chong CS, Low YHM, Sivakumar AI, and Gay KL (2007). Using a bee colony algorithm for neighborhood search in job shop scheduling problems. In *Proceedings of the 21st European Conference on Modeling and Simulation (ECMS '07)*, 459–465. ECMS.

Cicirello VA and Smith SF (2004). Wasp-like Agents for Distributed Factory Coordination. In *Autonomous Agents and Multi-Agent Systems*, 8(3): 237–266.

Couzin ID and Krause J (2003). Self-Organization and Collective Behavior in Vertebrates. volume 32 of *Advances in the Study of Behavior*, 1–75. Academic Press.

Couzin ID, Krause J, Franks NR, and Levin SA (2005). Effective leadership and decision-making in animal groups on the move. In *Nature*, 433(7025): 513–516.

Cramer JS (2003). The origins and development of the logit mode.

Cresswell KA, Satterthwaite WH, and Sword GA (2011). *Animal Migration: A Synthesis*, chapter Understanding the evolution of migration through empirical examples, 7–16. Oxford University Press, New York, NY, USA.

Crist TO, Guertin DS, Wiens JA, and Milne BT (1992). Animal movement in heterogeneous landscapes: an experiment with *Eleodes* beetles in shortgrass prairie. In *Functional Ecology*, 6(5): 536–544.

Crone EE and Schultz CB (2008). Old Models Explain New Observations Of Butterfly Movement At Patch Edges. In *Ecology*, 89(7): 2061–2067.

de Vries H and Biesmeijer JC (1998). Modelling collective foraging by means of individual behaviour rules in honey-bees. In *Behavioral Ecology and Sociobiology*, 44(2): 109–124.

Detrain C and Pasteels JM (1992). Caste polyethism and collective defense in the ant, *Pheidole pallidula*: the outcome of quantitative differences in recruitment. In *Behavioral Ecology and Sociobiology*, 29(6): 405–412.

Di Caro G and Dorigo M (1998). AntNet: Distributed Stigmergetic Control for Communications Networks. In *Journal of Artificial Intelligence Research*, 9(1): 317–365.

Di Caro G, Ducatelle F, and Gambardella LM (2005). AntHocNet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks. In *European Transactions on Telecommunications*, 16(5): 443–455.

Diwold K (2005). *The influence of dynamic environments on polyethism in response threshold models for social insects.* Bachelor's thesis, Institute of Cognitive Science, University of Osnabrück, Germany.

Diwold K, Aderhold A, Scheidler A, and Middendorf M (2011a). Performance Evaluation of Artificial Bee Colony Optimization and New Selection Schemes. In *Memetic Computing (in press)*.

Diwold K, Beekman M, and Middendorf M (2010). Bee Nest Site Selection as an Optimization Process. In *Proceedings of the 12th international conference on the synthesis and simulation of living systems (ALife XII)*, 626–633. The MIT Press.

Diwold K, Beekman M, and Middendorf M (2011b). Honeybee Optimisation – An Overview and a New Bee Inspired Optimisation Scheme. In LM Hiot, YS Ong, BK Panigrahi, Y Shi, and MH Lim (editors), *Handbook of Swarm Intelligence*, volume 8 of *Adaptation, Learning, and Optimization*, 295–327. Springer Berlin / Heidelberg.

Diwold K, Himmelbach D, Meier R, Baldauf C, and Middendorf M (2011c). Bonding as a swarm: applying bee nest-site selection behaviour to protein docking. In *Proceedings of the 13th annual conference on genetic and evolutionary computation (GECCO '11)*, 93–100. ACM.

Diwold K, Merkle D, and Middendorf M (2009a). Adapting to Dynamic Environments: Polyethism in Response Threshold Models for Social Insects. In *Advances in Complex Systems*, 12(3): 327–346.

Diwold K, Schaerf TM, Myerscough MR, Middendorf M, and Beekman M (2011d). Deciding on the wing: in-flight decision making and search space sampling in the red dwarf honeybee *Apis florea*. In *Swarm Intelligence*, 5(2): 121–141.

Diwold K, Scheidler A, and Middendorf M (2009b). The Effect of Spatial Organisation in Response Threshold Models for Social Insects. In *Proceedings of the European Conference on Complex Systems (ECCS '09)*. University of Warwick, UK.

Dorigo M (1992). *Ottimizzazione, apprendimento automatico, ed algoritmi basati su metafora naturale (Optimization, Learning and Natural Algorithms)*. Ph.D. thesis, Politecnico di Milano, Italy. In italian.

Dorigo M and Stützle T (2004). *Ant Colony Optimization*. MIT Press, Cambridge, MA, USA.

Dornhaus A, Kluegl 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. In *Behavioral Ecology*, 17(3): 336–344.

Dussutour A, Beekman M, Nicolis S, and Meyer B (2009). Noise improves collective decision-making by ants in dynamic environments. In *Proceedings of the Royal Society. Series B: Biological Sciences*, 276(1677): 4353–4361.

Eiben AE and Smith JE (2003). *Introduction to Evolutionary Computing*. Springer Berlin / Heidelberg.

Farooq M (2008). *Bee-Inspired Protocol Engineering: From Nature to Networks*. Natural Computation Series. Springer, Berlin / Heidelberg.

Fathian M and Amiri B (2008). A honeybee-mating approach for cluster analysis. In *The International Journal of Advanced Manufacturing Technology*, 38(7-8): 809–821.

Fernandez-Marin H, Zimmermann JK, and Wcislo WT (2003). Nest-founding in *Acromyrmex octospinosus* (Hymenoptera, Formicidae, Attini): demography and putative prophylactic behaviors. In *Insectes Sociaux*, 50(4): 304–308.

Fewell JH and Bertram SM (1999). Division of labor in a dynamic environment: response by honeybees (*Apis mellifera*) to graded changes in colony pollen stores. In *Behavioral Ecology and Sociobiology*, 46(3): 171–179.

Focardi S, Montanaro P, and Pecchioli E (2009). Adaptive Lévy Walks in Foraging Fallow Deer. In *PLoS ONE*, 4(8): e6587.

Franks NR, Dornhaus A, Fitzsimmons JP, and Stevens M (2003). Speed versus accuracy in collective decision making. In *Proceedings of the Royal Society. Series B: Biological Sciences*, 270(1532): 2457–2463.

Franks NR, Gomez N, Goss S, and Deneubourg JL (1991). The blind leading the blind in army ant raid patterns: Testing amodel of self-organization (*Hymenoptera: Formicidae*). In *Journal of Insect Behavior*, 4(5): 583–607.

Franks NR and Richardson T (2006). Teaching in tandem-running ants. In *Nature*, 439(7073): 153.

Fuchs S and Schade V (1994). Lower performance in honeybee colonies of uniform paternity. In *Apidologie*, 25(2): 155–168.

Gardner KE, Seeley TD, and Calderone NW (2008). Do honeybees have two discrete dances to advertise food sources? In *Animal Behaviour*, 75(4): 1291–1300.

Garnier S (2011). From Ants to Robots and Back: How Robotics Can Contribute to the Study of Collective Animal Behavior. In Y Meng and Y Jin (editors), *Bio-Inspired Self-Organizing Robotic Systems*, volume 355 of *Studies in Computational Intelligence*, 105–120. Springer Berlin / Heidelberg.

Gautrais J, Theraulaz G, Deneubourg JL, and Anderson C (2002). Emergent polyethism as a consequence of increased colony size in insect societies. In *Journal of Theoretical Biology*, 215(3): 363–373.

Giurfa M, Vorobyev M, Kevan P, and Menzel R (1996). Detection of coloured stimuli by honeybees: minimum visual angles and receptor specific contrasts. In *Journal of Comparative Physiology A*, 178(5): 699–709.

Gordon N, Wagner IA, and Brucks AM (2003). Discrete Bee Dance Algorithms for Pattern Formation on a Grid. In *Proceedings of the 2003 IEEE/WIC International Conference on Intelligent Agent Technology (IAT '03)*, 545–549. IEEE Computer Society.

Goss S, Aron S, Deneubourg JL, and Pasteels JM (1989). Self-organized short-cuts in the argentine ant. In *Naturwissenschaften*, 76(12): 579–581.

Greenwood GW and Abbass HA (2007). A new local search algorithm for continuous spaces based on army ant swarm raids. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '07)*, 1097–1102. IEEE Computer Society.

Grünbaum D (1998). Schooling as a strategy for taxis in a noisy environment. In *Evolutionary Ecology*, 12(5): 503–522.

Grüter C and Farina WM (2009). The honeybee waggle dance: can we follow the steps? In *Trends in Ecology and Evolution*, 24(5): 242–247.

Grüter C and Ratnieks FLW (2011). Honeybee foragers increase the use of waggle dance information when private information becomes unrewarding. In *Animal Behaviour*, 81(5): 949–954.

Häckel S and Dippold P (2009). The bee colony-inspired algorithm (BCiA): a two-stage approach for solving the vehicle routing problem with time windows. In *Proceedings of the 11th Annual Conference on Genetic and evolutionary computation (GECCO '09)*, 25–32. ACM.

Haddad OB, Afshar A, and Marino MA (2008). Design-Operation of Multi-Hydropower Reservoirs: HBMO Approach. In *Water Resources Management*, 22(12): 1709–1722.

Halloy J, Sempo G, Caprari G, Rivault C, Asadpour M, Tache F, Said I, Durier V, Canonge S, Amé JM, Detrain C, Correll N, Martinoli A, Mondada F, Siegwart R, and Deneubourg JL (2007). Social Integration of Robots into Groups of Cockroaches to Control Self-Organized Choices. In *Science*, 318(5853): 1155–1158.

Halperin I, Ma B, Wolfson H, and Nussinov R (2002). Principles of docking: an overview of search algorithms and a guide to scoring functions. In *Proteins*, 47(4): 409–443.

Handl J and Meyer B (2007). Ant-based and swarm-based clustering. In *Swarm Intelligence*, 1(2): 95–113.

HELBING D, JOHANSSON A, AND AL-ABIDEEN HZ (2007). Dynamics of crowd disasters: An empirical study. In *Physical Review E*, 75(4): 046109.

HIMMELBACH D (2011). *Entwicklung eines Bienenalgorithmus zur Lösung des Protein-Docking-Problems*. Master's thesis, Universität Leipzig.

HUANG ZY AND ROBINSON GE (1992). Honeybee colony integration: worker-worker interactions mediate hormonally regulated plasticity in division of labor. In *Proceedings of the National Academy of Sciences*, 89(24): 11726–11729.

HUANG ZY AND ROBINSON GE (1996). Regulation of honey bee division of labor by colony age demography. In *Behavioral Ecology and Sociobiology*, 39(3): 147–158.

IOANNOU CC, COUZIN ID, JAMES R, CROFT DP, AND KRAUSE J (2011). *Fish Cognition and Behavior*, chapter Social Organisation and Information Transfer in Schooling Fish, 217–239. Wiley-Blackwell, Hoboken, NJ, USA.

JANDT JM AND DORNHAUS A (2009). Spatial organization and division of labour in the bumblebee *Bombus impatiens*. In *Animal Behaviour*, 77(3): 641–651.

JANMAAT AF AND WINSTON ML (2000). The influence of pollen storage area and Varroa jacobsoni Oudemans parasitism on temporal caste structure in honeyb bees (*Apis mellifera* L.). In *Insectes Sociaux*, 47(2): 177–182.

JANSON S, MERKLE D, AND MIDDENDORF M (2008). Molecular docking with multi-objective Particle Swarm Optimization. In *Applied Soft Computing*, 8(1): 666–675.

JANSON S, MIDDENDORF M, AND BEEKMAN M (2005). Honeybee swarms: how do scouts guide a swarm of uninformed bees? In *Animal Behaviour*, 70(1): 349–358.

JANSON S, MIDDENDORF M, AND BEEKMAN M (2007). Searching for a new home – scouting behavior of honeybee swarms. In *Behavioral Ecology*, 18(2): 384–392.

JEANNE RL AND NORDHEIM EV (1996). Productivity in a social wasp: per capita output increases with swarmsize. In *Behavioral Ecology*, 7(1): 43–48.

JEANSON R, FEWELL JH, GORELICK R, AND BERTRAM SM (2007). Emergence of increased division of labor as a function of group size. In *Behavioral Ecology and Sociobiology*, 62(2): 289–298.

JOHNSON BR (2009). Pattern formation on the combs of honeybees: increasing fitness by coupling self-organization with templates. In *Proceedings of the Royal Society. Series B: Biological Sciences*, 276(1655): 255–261.

JONES G, WILLETT P, AND GLEN RC (1995). Molecular regcognition of receptor sites using a genetic algorithm with a description of desolvation. In *Journal of Molecular Biology*, 245(1): 43–53.

JONES JC, MYERSCOUGH MR, GRAHAM S, AND OLDROYD BP (2004). Honey bee nest thermoregulation: diversity promotes stability. In *Science*, 305(5682): 402–404.

JUNG SH (2003). Queen-bee evolution for genetic algorithms. In *Electronics Letters*, 39(6): 575–576.

KAMPSTRA P (2008). Beanplot: A Boxplot Alternative for Visual Comparison of Distributions. In *Journal of Statistical Software*, 28(1): 1–9.

KARABOGA D (2005). An Idea Based On Honey Bee Swarm for Numerical Optimization. Technical Report TR06, Erciyes University, Engineering Faculty, Computer Engineering Department.

KARABOGA D AND AKAY B (2009). A survey: algorithms simulating bee swarm intelligence. In *Artificial Intelligence Review*, 31(1-4): 61–85.

KARABOGA D, AKAY B, AND OZTURK C (2007). Artificial Bee Colony (ABC) Optimization Algorithm for Training Feed-Forward Neural Networks. In *Proceedings of the 4th international conference on Modeling Decisions for Artificial Intelligence (MDAI '07)*, 318–329. Springer Berlin / Heidelberg.

KARABOGA D AND BASTURK B (2007a). Artificial Bee Colony (ABC) Optimization Algorithm for Solving Constrained Optimization Problems. In P Melin, O Castillo, L Aguilar, J Kacprzyk, and W Pedrycz (editors), *Foundations of Fuzzy Logic and Soft Computing*, volume 4529 of *Lecture Notes in Computer Science*, 789–798. Springer Berlin / Heidelberg.

KARABOGA D AND BASTURK B (2007b). A powerful and efficient algorithm for numerical function optimization: artifcial bee colony (ABC) algorithm. In *Journal of Global Optimization*, 39(3): 459–471.

KARABOGA D AND BASTURK B (2008). On the performance of artifcial bee colony (ABC) algorithm. In *Applied Soft Computing*, 8(1): 687–697.

KARABOGA D AND OZTURK C (2011). A novel clustering approach: Artificial Bee Colony (ABC) algorithm. In *Applied Soft Computing*, 11(1): 652–657.

KARCI A (2004). Imitation of Bee Reproduction as a Crossover Operator in Genetic Algorithms. In C Zhang, H W Guesgen, and WK Yeap (editors), *PRICAI 2004: Trends* *in Artificial Intelligence*, volume 3157 of *Lecture Notes in Computer Science*, 1015–1016. Springer Berlin / Heidelberg.

Karsai I and Wenzel JW (2000). Organization and regulation of nest construction behavior in *Metapolybia* wasps. In *Journal of Insect Behaviour*, 13(1): 111–140.

Kennedy J and Eberhart R (1995). Particle swarm optimization. In *Proceedings of IEEE International Conference on Neural Networks (ICNN '95)*, volume 4, 1942–1948. IEEE.

Korb O, Stützle T, and Exner T (2006). PLANTS: Application of Ant Colony Optimization to Structure-Based Drug Design. In MD et al (editor), *Ant Colony Optimization and Swarm Intelligence*, volume 4150 of *Lecture Notes in Computer Science*, 247–258. Springer Berlin / Heidelberg.

Kramer DL and McLaughlin RL (2001). The behavioral ecology of intermittent locomotion. In *Integrative and Comparative Biology*, 41(2): 137–153.

Krause J and Ruxton GD (2002). *Living in groups.* Oxford series in Ecology and Evolution. Oxford University Press, New York, NY, USA.

Krieger MJB, Billeter JB, and Keller L (2000). Ant–like task allocation and recruitment in cooperative robots. In *Nature*, 406(6799): 992–995.

Kugler H (1933). Blütenökologische Untersuchungen mit Hummeln VI. In *Planta*, 19(4): 781–789.

Kühnholz S and Seeley TD (1997). The control of water collection in honey bee colonies. In *Behavioral Ecology and Sociobiology*, 41(6): 407–422.

Labella TH, Dorigo M, and Deneubourg JL (2006). Division of labor in a group of robots inspired by ants' foraging behavior. In *ACM: Transactions on Autonomous and Adaptive Systems*, 1(1): 4–25.

Langridge EA, Sendova-Franks AB, and Franks NR (2008). How experienced individuals contribute to an improvement in collective performance in ants. In *Behavioral Ecology and Sociobiology*, 62(3): 447–456.

Latty T, Duncan M, and Beekman M (2009). High bee traffic disrupts transfer of directional information in flying honeybee swarms. In *Animal Behaviour*, 78(1): 117–121.

Latty T, Ramsch K, Ito K, Nakagaki T, Sumpter DJT, Middendorf M, and Beekman M (2011). Structure and formation of ant transportation networks. In *Journal of the Royal Society Interface*, 8(62): 1298–1306.

Lemmens N, de Jong S, Tuyls K, and Nowé A (2008). Bee Behaviour in Multi-agent Systems. In K Tuyls, A Nowe, Z Guessoum, and D Kudenko (editors), *Adaptive Agents and Multi-Agent Systems III. Adaptation and Multi-Agent Learning*, volume 4865 of *Lecture Notes in Computer Science*, 145–156. Springer Berlin / Heidelberg.

Lihoreau M, Chittka L, and Raine NE (2010). Travel optimization by foraging bumblebees through readjustments of traplines after discovery of new feeding locations. In *The American Naturalist*, 176(6): 744–757.

Lindauer M (1955). Schwarmbienen auf Wohnungssuche. In *Zeitschrift für vergleichende Physiologie*, 37(4): 263–324.

Liu B, Chen H, Huang H, Hwang S, and Ho S (2005). Flexible protein-ligand docking using Particle Swarm Optimization. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '05)*, volume 1, 251–258. IEEE.

Lu X and Zhou Y (2008). A Novel Global Convergence Algorithm: Bee Collecting Pollen Algorithm. In DS Huang, D Wunsch, D Levine, and KH Jo (editors), *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence*, volume 5227 of *Lecture Notes in Computer Science*, 518–525. Springer Berlin / Heidelberg.

Lucic P and Teodorovic D (2001). Bee system: Modeling combinatorial optimization transportation engineering problems by swarm intelligence. In *Preprints of the TRISTAN IV Triennial Symposium on Transportation Analysis*, 441–445. São Miguel, Portugal.

Lucic P and Teodorovic D (2002). Transportation modeling: an artificial life approach. In *Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence (ICTAI '02)*, 216–223. IEEE.

Lucic P and Teodorovic D (2003). Computing with Bees: Attacking Complex Transportation Engineering Problems. In *International Journal on Artificial Intelligence Tools*, 12(3): 375–394.

Makinson JC, Oldroyd BP, Schaerf TM, Wattanachaiyingcharoen W, and Beekman M (2011). Moving home: Nest-site selection in the Red Dwarf Honeybee (*Apis florea*). In *Behavioral Ecology and Sociobiology*, 65(5): 945–958.

MARINAKIS Y AND MARINAKI M (2010). Bumble Bees Mating Optimization Algorithm for the Vehicle Routing Problem. In LM Hiot, YS Ong, BK Panigrahi, Y Shi, and MH Lim (editors), *Handbook of Swarm Intelligence*, volume 8 of *Adaptation, Learning, and Optimization*, 347–369. Springer Berlin / Heidelberg.

MARINAKIS Y, MARINAKI M, AND DOUNIAS G (2009a). Honey Bees Mating Optimization algorithm for large scale vehicle routing problems. In *Natural Computing*, 9(1): 5–27.

MARINAKIS Y, MARINAKI M, AND MATSATSINIS N (2008). A Hybrid Clustering Algorithm Based on Honey Bees Mating Optimization and Greedy Randomized Adaptive Search Procedure. In V Maniezzo, R Battiti, and JP Watson (editors), *Learning and Intelligent Optimization*, volume 5313 of *Lecture Notes in Computer Science*, 138–152. Springer Berlin / Heidelberg.

MARINAKIS Y, MARINAKI M, AND MATSATSINIS N (2009b). A Hybrid Bumble Bees Mating Optimization - GRASP Algorithm for Clustering. In E Corchado, X Wu, E Oja, A Herrero, and B Baruque (editors), *Hybrid Artificial Intelligence Systems*, volume 5572 of *Lecture Notes in Computer Science*, 549–556. Springer Berlin / Heidelberg.

MARINAKIS Y, MARINAKI M, AND MATSATSINIS N (2010). A Bumble Bees Mating Optimization Algorithm for Global Unconstrained Optimization Problems. In J González, D Pelta, C Cruz, G Terrazas, and N Krasnogor (editors), *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, volume 284 of *Studies in Computational Intelligence*, 305–318. Springer Berlin / Heidelberg.

MARSHALL JAR, BOGACY R, DORNHAUS A, PLANQUÉ R, KOVACS T, AND FRANKS NR (2009). On optimal decision-making in brains and social insect colonies. In *Journal of the Royal Society Interface*, 6(40): 1065–1074.

MATSUMOTO M AND NISHIMURA T (1998). Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. In *ACM: Transactions on Modeling and Computer Simulation*, 8(1): 3–30.

MATTILA HR, BURKE KM, AND SEELEY TD (2008). Genetic diversity within honeybee colonies increases signal production by waggle-dancing foragers. In *Proceedings of the Royal Society. Series B: Biological Sciences*, 275(1636): 809–816.

MATTILA HR AND SEELEY TD (2007). Genetic diversity in honey bee colonies enhances productivity and fitness. In *Science*, 317(5836): 362–364.

MAUTZ D (1971). Der Kommunikationseffekt der Schwänzeltänze bei *Apis mellifera carnica.* In *Zeitschrift für vergleichende Physiologie*, 72(2): 192–220.

MEIER R, PIPPEL M, BRANDT F, SIPPL W, AND BALDAUF C (2010). PARADOCKS: A Framework for Molecular Docking with Population-Based Metaheuristics. In *Journal of Chemical Information and Modeling*, 50(5): 879–889.

MERKLE D AND MIDDENDORF M (2004). Dynamic polyethism and competition for tasks in threshold reinforcement models of social insects. In *Adaptive Behavior*, 12(3-4): 251–262.

MOHAN S AND BABU KSJ (2010). Optimal Water Distribution Network Design with Honey-Bee Mating Optimization. In *Journal of Computing in Civil Engineering*, 24(1): 117–126.

MONMARCHÉ N, VENTURINI G, AND SLIMANE M (2000). On how Pachycondyla apicalis ants suggest a new search algorithm. In *Future Generation Computer Systems*, 16(9): 937–946.

MUEGGE I (2006). PMF scoring revisited. In *Journal of Medicinal Chemistry*, 49(20): 5895–5902.

MYERSCOUGH MR (2003). Dancing for a decision: a matrix model for nest-site choice by honeybees. In *Proceedings of the Royal Society. Series B: Biological Sciences*, 270(1515): 577–582.

NAKRANI S AND TOVEY C (2004). On Honey Bees and Dynamic Server Allocation in Internet Hosting Centers. In *Adaptive Behavior - Animals, Animats, Software Agents, Robots, Adaptive Systems*, 12(3-4): 223–240.

NEILL W (1979). Mechanisms of fish distribution in heterothermal environments. In *Integrative and Comparative Biology*, 19(1): 305–317.

NEVAI AL, PASSINO KM, AND SRINIVASAN P (2010). Stability of choice in the honey bee nest-site selection process. In *Journal of Theoretical Biology*, 263(1): 93–107.

OLAGUE G AND PUENTE C (2006). The Honeybee Search Algorithm for Three-Dimensional Reconstruction. In F Rothlauf, J Branke, S Cagnoni, E Costa, C Cotta, R Drechsler, E Lutton, P Machado, J Moore, J Romero, G Smith, G Squillero, and H Takagi (editors), *Applications of Evolutionary Computing*, volume 3907 of *Lecture Notes in Computer Science*, 427–437. Springer Berlin / Heidelberg.

OLDROYD BP AND FEWELL JH (2007). Genetic diversity promotes homeostasis in insect colonies. In *Trends in Ecology and Evolution*, 22(8): 408–413.

OLDROYD BP, GLOAG RS, EVEN N, WATTANACHAIYINGCHAROEN W, AND BEEKMAN M (2008). Nest-site selection in the open-nesting honey bee *Apis florea*. In *Behavioral Ecology and Sociobiology*, 62(10): 1643–1653.

OLDROYD BP, SMOLENSKI A, LAWLER S, ESTOUP A, AND CROZIER R (1995). Colony aggregations in *Apis mellifera* L. In *Apidologie*, 26(2): 119–130.

PALMER KA AND OLDROYD BP (2000). Evolution of multiple mating in the genus *Apis*. In *Apidologie*, 31(2): 235–248.

PANKIW T AND PAGE RE (1999). The effect of genotype, age, sex, and caste on response thresholds to sucrose and foraging behavior of honey bees (*Apis mellifera* L.). In *Journal of Comparative Physiology A*, 185(2): 207–213.

PASSINO KM AND SEELEY TD (2006). Modeling and analysis of nest-site selection by honeybee swarms: the speed and accuracy trade-off. In *Behavioral Ecology and Sociobiology*, 59(3): 427–442.

PERDRIAU BS AND MYERSCOUGH MR (2007). Making good choices with variable information: a stochastic model for nest-site selection by honeybees. In *Biology Letters*, 3(2): 140–143.

PERRON P (1988). Trends and random walks in macroeconomic time series: Further evidence from a new approach. In *Journal of Economic Dynamics and Control*, 12(2-3): 297–332.

PHAM DT, DARWISH AH, AND ELDUKHRI EE (2009). Optimisation of a fuzzy logic controller using the Bees Algorithm. In *International Journal of Computer Aided Engineering and Technology*, 1(2): 250–264.

PHAM DT AND GHANBARZADEH A (2007). Multi-Objective Optimisation using the Bees Algorithm. In *Proceedings of the Innovative Production Machines and Systems Conference (IPROMS '07)*. Whittles Publishing.

PHAM DT, GHANBARZADEH A, KOC E, AND OTRI S (2006a). Application of the Bees Algorithm to the Training of Radial Basis Function Networks for Control Chart Pattern Recognition. In *Proceedings of the 5th CIRP International Seminar on Intelligent Computation in Manufacturing Engineering (CIRP ICME '06)*, 7110–716. Ischia, Italy.

PHAM DT, GHANBARZADEH A, KOC E, OTRI S, RAHIM S, AND ZAIDI M (2006b). The Bees Algorithm – A Novel Tool for Complex Optimisation Problems. In *Proceedings of the Innovative Production Machines and Systems Conference (IPROMS '06)*, 454–461. Elsevier Science.

PHAM DT, KOC E, GHANBARZADEH A, AND OTRI S (2006c). Optimisation of the Weights of Multi-Layered Perceptrons Using the Bees Algorithm. In *Proceedings of the 5th International Symposium on Intelligent Manufacturing Systems (IMS '06)*. University of Sakarya, Turkey.

PHAM DT, PHAM QT, GHANBARZADEH A, AND CASTELLANI M (2008). Dynamic Optimisation of Chemical Engineering Processes Using the Bees Algorithm. In *Proceedings of the 17th IFAC World Congress (IFAC '08)*, 6100–6105. Elsevier Science.

PHAM DT, SOROKA AJ, KOC E, GHANBARZADEH A, OTRI S, AND PACKIANATHER M (2006d). Optimising Neural Networks for Identification of Wood Defects Using the Bees Algorithm. In *Proceedings of the 4th International IEEE Conference on Industrial Informatics (INDIN '06)*, 1346 –1351. IEEE.

POLI R, KENNEDY J, AND BLACKWELL T (2007). Particle swarm optimization: An overview. In *Swarm Intelligence*, 1(1): 33–57.

PRATT SC, MALLON E, SUMPTER DJT, AND FRANKS NR (2002). Quorum sensing, recruitment, and collective decision-making during colony emigration by the ant *Leptothorax albipennis*. In *Behavioral Ecology and Sociobiology*, 52(2): 117–127.

REEBS SG (2000). Can a minority of informed leaders determine the foraging movements of a fish shoal? In *Animal Behaviour*, 59(2): 403–409.

REYNOLDS AM (2008). How Many Animals Really Do the Lévy Walk? Comment. In *Ecology*, 89(8): 2347–2351.

REYNOLDS AM, SMITH AD, MENZEL R, GREGGERS U, REYNOLDS DR, AND RILEY JR (2007). Displaced Honey Bees perform optimal scale-free search flights. In *Ecology*, 88(8): 1955–1961.

REYNOLDS AM, SWAIN JL, SMITH AD, MARTIN A, AND OSBORNE J (2009). Honeybees use a Lévy flight search strategy and odour-mediated anemotaxis to relocate food sources. In *Behavioral Ecology and Sociobiology*, 64(1): 115–123.

REYNOLDS CW (1987). Flocks, Herds, and Schools: A Distributed Behavioral Model. In *Computer Graphics*, 21(4): 25–34.

RIBBANDS CR (1949). The Foraging Method of Individual Honey-Bees. In *Journal of Animal Ecology*, 18(1): 47–66.

RICHARDSON TO, CHRISTENSEN K, FRANKS NR, JENSEN HJ, AND SENDOVA-FRANKS AB (2011). Ants in a Labyrinth: A Statistical Mechanics Approach to the Division of Labour. In *PLoS ONE*, 6(4): e18416.

Riley JR, Greggers U, Smith AD, Reynolds DR, and Menzel R (2005). The flight paths of honeybees recruited by the waggle dance. In *Nature*, 435(7039): 205–207.

Robinson GE (1987). Regulation of honey bee age polyethism by juvenile hormone. In *Behavioral Ecology and Sociobiology*, 20(5): 329–338.

Robinson GE (1992). Regulation of Division of Labor in Insect Societies. In *Annual Review of Entomology*, 37: 637–665.

Robinson GE and Page RE (1989). Genetic basis for division of labor in an insect society. In BM D and RE Page (editors), *The genetics of social evolution*, 61–80. Westview Press, Bolder, CO.

Robinson GE, Page RE, Strambi C, and Strambi A (1989). Hormonal and genetic control of behavioral integrationin honey bee colonies. In *Science*, 246(4926): 109–12.

Russell R (1997). Heat trails as short-lived navigational markers for mobile robots. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation (ICRA '97)*, volume 4, 3534–3539. IEEE.

Sato T and Hagiwara M (1997). Bee System: finding solution by a concentrated search. In *Proceedings of the 1997 IEEE International Conference on Systems, Man, and Cybernetics (SMC '97)*, volume 4, 3954–3959. IEEE.

Scheiner R, Page RE, and Erber J (2004). Sucrose responsiveness and behavioral plasticity in honey bees (*Apis mellifera*). In *Apidologie*, 35(2): 133–142.

Schmickl T and Crailsheim K (2008). Trophallaxis within a robotic swarm: bio-inspired communication among robots in a swarm. In *Autonomous Robots*, 25(1-2): 171–188.

Schneider SS (1995). Swarm movement patterns inferred from waggle dance activity of the neotropical African honey bee in Costa Rica. In *Apidologie*, 26(5): 395–406.

Schultz KM, Passino KM, and Seeley TD (2008). The mechanism of flight guidance in honeybee swarms: subtle guides or streaker bees? In *Journal of Experimental Biology*, 211(20): 3287–3295.

Seeley TD (2003). Consensus building during nest-site selection in honey bee swarms: the expiration of dissent. In *Behavioral Ecology and Sociobiology*, 53(6): 417–424.

Seeley TD (2010). *Honeybee democracy.* Princeton University Press, Princton, NJ, USA.

Seeley TD and Buhrman SC (1999). Group decision making in swarms of honey bees. In *Behavioral Ecology and Sociobiology*, 45(1): 19–31.

Seeley TD and Buhrman SC (2001). Nest-site selection in honey bees: how well do swarms implement the "best-of-n" decision rule. In *Behavioral Ecology and Sociobiology*, 49(5): 416–427.

Seeley TD, Camazine S, and Sneyd J (1991). Collective decision-making in honey bees: how colonies choose among nectar sources. In *Behavioral Ecology and Sociobiology*, 28(4): 277–290.

Seeley TD, Kleinhenz M, Bujok B, and Tautz J (2003). Thorough warm-up before take-off in honey bee swarms. In *Naturwissenschaften*, 90(6): 256–260.

Seeley TD and Kolmes SA (1991). Age Polyethism for Hive Duties in Honey Bees – Illusion or Reality? In *Ethology*, 87(3-4): 284–297.

Seeley TD, Mikheyev AS, and Pagano GJ (2000). Dancing bees tune both duration and rate of waggle-run production in relation to nectar-source profitability. In *Journal of Comparative Physiology A*, 186(9): 813–819.

Seeley TD and Morse RA (1978). Nest site selection by the honey bee, *Apis mellifera*. In *Insectes Sociaux*, 25(4): 323–337.

Seeley TD, Morse RA, and K VP (1979). The natural history of the flight of honey bee swarms. In *Psyche*, 86(2-3): 103–114.

Seeley TD, Seeley RH, and Akratanakul P (1982). Colony defense strategies of the honeybees in Thailand. In *Ecological Monographs*, 52(1): 43–63.

Seeley TD and Visscher PK (1985). Survival of honeybees in cold climates: the critical timing of colony growth and reproduction. In *Ecological Entomology*, 10(1): 81–88.

Seeley TD and Visscher PK (2003). Choosing a home: how the scouts in a honey bee swarm perceive the completion of their group decision making. In *Behavioral Ecology and Sociobiology*, 54(5): 511–520.

Seeley TD and Visscher PK (2004). Quorum sensing during nest-site selection by honeybee swarms. In *Behavioral Ecology and Sociobiology*, 56(6): 594–601.

Seeley TD and Visscher PK (2008). Sensory coding of nest-site value in honeybee swarms. In *Journal of Experimental Biology*, 211(23): 3691–3697.

Sendova-Franks AB and Franks NR (1993). Task allocation in ant colonies within variable environments (a study of temporal polyethism: experimental). In *Bulletin of Mathematical Biology*, 55(1): 75–96.

Sendova-Franks AB and Franks NR (1994). Social resilience in individual worker ants and its role in division of labour. In *Proceedings of the Royal Society. Series B: Biological Sciences*, 256(1347): 305–309.

Sherman G and Visscher PK (2002). Honeybee colonies achieve fitness through dancing. In *Nature*, 419(6910): 920–922.

Simpson SJ and Sword GA (2010). Evolving migration. In *Proceedings of the National Academy of Sciences*, 107(39): 16753–16754.

Srinivasan MV (2011). Honeybees as a Model for the Study of Visually Guided Flight, Navigation, and Biologically Inspired Robotics. In *Physiological Reviews*, 91(2): 413–460.

Srinivasan MV and Reinhard J (2009). *Food Exploitation By Social Insects: Ecological, Behavioral, and Theoretical Approaches*, chapter The Role of Scents in Honey Bee Foraging and Recruitment, 165–182. CRC Press, Boca Raton, FL, USA.

Subbotin SA and Oleinik AA (2009). Multiagent Optimization based on the bee-colony method. In *Cybernetics and Systems Analysis*, 45(2): 177–186.

Sumpter DJT (2010). *Collective Animal Behavior*. Princeton University Press, Princton, NJ, USA.

Sumpter DJT and Pratt SC (2003). A modelling framework for understanding social insect foraging. In *Behavioral Ecology and Sociobiology*, 53(3): 131–144.

Sumpter DJT and Pratt SC (2009). Quorum responses and consensus decision making. In *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 364(1518): 743–753.

Swaney W, Kendal J, Capon H, Brown C, and Laland KN (2001). Familiarity facilitates social learning of foraging behaviour in the guppy. In *Animal Behaviour*, 62(3): 591–598.

Tautz J and Rohrseitz K (1998). What attracts honeybees to a waggle dancer? In *Journal of Comparative Physiology A*, 183(5): 661–667.

Tautz J, Rohrseitz K, and Sandeman DC (1996). One-strided waggle dance in bees. In *Nature*, 382(6586): 32.

Teodorovic D and Dell'Orco M (2005). Bee colony optimziation - a cooperative learning approach to complex transportation problems. In *Advanced OR and AI Methods in Transportation. Proceedings of the 10th Jubilee Meeting of the Euro Working Group on Transportation (EWGT '05)*, 51–60. Poznan University of Technology Publishers.

Teodorovic D and Dell'Orco M (2008). Mitigating traffic congestion: solving the ride-matching problem by bee colony optimization. In *Transportation Planning and Technology*, 31(2): 135–152.

Teodorovic D, Lucic P, Markovic G, and Dell'Orco M (2006). Bee Colony Optimization: Principles and Applications. In *Proceedings of the 8th Seminar on Neural Network Applications in Electrical Engineering (NEUREL '06)*, 151–156. IEEE.

Theraulaz G and Bonabeau E (1999). A Brief History of Stigmergy. In *Artificial Life*, 5(2): 97–116.

Theraulaz G, Bonabeau E, and Denuebourg JL (1998). Response threshold reinforcements and division of labour in insect societies. In *Proceedings of the Royal Society. Series B: Biological Sciences*, 265(1393): 327–332.

Thom C, Gilley DC, Hooper J, and Esch HE (2007). The Scent of the Waggle Dance. In *PLoS Biology*, 5(9): e228.

Tofilski A (2002). Influence of age polyethism on longevity of workers in social insects. In *Behavioral Ecology and Sociobiology*, 51(3): 234–237.

Trianni V and Dorigo M (2006). Self-organisation and communication in groups of simulated and physical robots. In *Biological Cybernetics*, 95(3): 213–231.

Turing AM (1952). The Chemical Basis of Morphogenesis. In *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 237(641): 37–72.

Vaughan RT, Sty K, Sukhatme GS, and Mataric MJ (2002). LOST: Localization-Space Trails for Robot Teams. In *IEEE Transactions on Robotics and Automation*, 18(5): 796–812.

Vicsek T and Zafiris A (2010). Collective motion. In *ArXiv e-prints*, arXiv:1010.5017.

Visscher PK (2007). Group decision making in nest-site selection among social insects. In *Annual Review Entomology*, 52: 255–275.

Visscher PK and Seeley TD (1982). Foraging strategy of honeybee colonies in a temperate deciduous forest. In *Ecology*, 63(6): 1790–1801.

Visscher PK and Seeley TD (2007). Coordinating a group departure: who produces the piping signals on honeybee swarms? In *Behavioral Ecology and Sociobiology*, 61(10): 1615–1621.

Viswanathan GM, Afanasyev V, Buldyrev SV, Murphy EJ, Prince PA, and Stanley HE (1996). Lévy flight search patterns of wandering albatrosses. In *Nature*, 381(6581): 413–415.

von Frisch K (1967). *The dance language and orientation of bees.* Harvard University Press, Cambridge, MA, USA.

von Neumann J (1966). *Theory of Self-Reproducing Automata.* University of Illinois Press, Champaign, IL, USA.

Waibel M, Floreano D, Magnenat S, and Keller L (2006). Division of labour and colony efficiency in social insects: effects of interactions between genetic architecture, colony kin structure and rate of perturbations. In *Proceedings of the Royal Society. Series B: Biological Sciences*, 273(1595): 1815–1823.

Wang R, Fang X, Lu Y, and Wang S (2004). The PDBbind database: Collection of binding affinities for protein-ligand complexes with known three-dimensional structures. In *Journal of Medicinal Chemistry*, 47(12): 2977–2980.

Wedde HF and Farooq M (2006). A comprehensive review of nature inspired routing algorithms for fixed telecommunication networks. In *Journal of Systems Architecture*, 52(8): 461–484.

Wedde HF, Farooq M, Pannenbaecker T, Vogel B, Mueller C, Meth J, and Jeruschkat R (2005). BeeAdHoc: an energy efficient routing algorithm for mobile ad hoc networks inspired by bee behavior. In *Proceedings of the 2005 conference on Genetic and evolutionary computation (GECCO '05)*, 153–160. ACM.

Wedde HF, Farooq M, and Zhang Y (2004). BeeHive: An Efficient Fault-Tolerant Routing Algorithm Inspired by Honey Bee Behavior. In M Dorigo, M Birattari, C Blum, L Gambardella, F Mondada, and T StÃijtzle (editors), *Ant Colony Optimization and Swarm Intelligence*, volume 3172 of *Lecture Notes in Computer Science*, 83–94. Springer Berlin / Heidelberg.

Weidenmüller A and Seeley TD (1999). Imprecision in waggle dances of the honeybee (Apis mellifera) for nearby food sources: error or adaptation? In *Behavioral Ecology and Sociobiology*, 46(3): 190–199.

WILSON EO (1971). *The Insect Societies.* Harvard University Press, Cambridge, MA, USA.

WILSON EO (1984). The relation between caste ratios and division of labour in the ant genus *Pheidole* (Hymenoptera: Formicidae). In *Behavioral Ecology and Sociobiology*, 16(1): 89–98.

WILSON EO (2001). *Success, Dominance, and the Superorganism: The Case of the Social Insects.* University of California Press, Berkeley, CA, USA.

WILSON EO, FEWELL JH, AND GADAU J (editors) (2009). *Organization of Insect Societies: From Genome to Sociocomplexity.* Harvard University Press, Cambridge, MA, USA.

WILSON EO AND HÖLLDOBLER B (2005). Eusociality: Origin and consequences. In *Proceedings of the National Academy of Sciences*, 102(38): 13367–13371.

WINSTON ML (1987). *The Biology of the Honey Bee.* Harvard University Press, Cambridge, MA, USA.

WITHERS GS, E FS, AND ROBINSON GE (1993). Selective neuroanatomical plasticity and division of labour in the honeybee. In *Nature*, 364(6434): 238–240.

WONG LP, LOW MYH, AND CHONG CS (2008). A Bee Colony Optimization Algorithm for Traveling Salesman Problem. In *Proceedings of the Asia International Conference on Modelling and Simulation (AMS '08)*, volume 0, 818–823. IEEE Computer Society.

WONG LP, LOW MYH, AND CHONG CS (2009). An Efficient Bee Colony Optimization Algorithm for Traveling Salesman Problem using Frequency-based Pruning. In *Proceedings of the 7th IEEE International Conference on Industrial Informatics (INDIN '09)*, 775–782. IEEE.

WONGSIRI S, LEKPRAYOON C, THAPA R, THIRAKUPT K, RINDERER TE, SYLVESTER HA, OLDROYD BP, AND BOONCHAM U (1996). Comparative biology of *Apis andreniformis* and *Apis florea* in Thailand. In *Bee World*, 77(4): 23–35.

WOOLEY JC AND LIN HS (editors) (2005). *Catalyzing Inquiry at the Interface of Computing and Biology*, chapter Biological Inspiration for Computing, 247–297. National Academies Press, Washington, DC, USA, 1 edition.

WOYCIECHOWSKI M AND MOROŃ D (2009). Life expectancy and onset of foraging in the honeybee (*Apis mellifera*). In *Insectes Sociaux*, 56(2): 193–201.

YANG XS (2005). Engineering Optimizations via Nature-Inspired Virtual Bee Algorithms. In J Mira and J Alvarez (editors), *Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach*, volume 3562 of *Lecture Notes in Computer Science*, 317–323. Springer Berlin / Heidelberg.

ZHANG D, XIE G, YU J, AND WANG L (2007). Adaptive task assignment for multiple mobile robots via swarm intelligence approach. In *Robotics and Autonomous Systems*, 55(7): 572–588.

ZOLLNER PA AND LIMA SL (1999). Search strategies for landscape-level interpatch movements. In *Ecology*, 80(3): 1019–1030.

Dissertationsbezogene bibliographische Daten

Diwold K, Schaerf TM, Myerscough MR, Middendorf M, and Beekman M (2011d). Deciding on the wing: in-flight decision making and search space sampling in the red dwarf honeybee *Apis florea*. In *Swarm Intelligence*, 5(2): 121–141.

Diwold K, Himmelbach D, Meier R, Baldauf C, and Middendorf M (2011c). Bonding as a swarm: applying bee nest-site selection behaviour to protein docking. In *Proceedings of the 13th annual conference on genetic and evolutionary computation (GECCO '11)*, 93–100. ACM.

Diwold K, Beekman M, and Middendorf M (2011b). Honeybee Optimisation – An Overview and a New Bee Inspired Optimisation Scheme. In LM Hiot, YS Ong, BK Panigrahi, Y Shi, and MH Lim (editors), *Handbook of Swarm Intelligence*, volume 8 of *Adaptation, Learning, and Optimization*, 295–327. Springer Berlin / Heidelberg.

Diwold K, Beekman M, and Middendorf M (2010). Bee Nest Site Selection as an Optimization Process. In *Proceedings of the 12th international conference on the synthesis and simulation of living systems (ALife XII)*, 626–633. The MIT Press.

Diwold K, Scheidler A, and Middendorf M (2009b). The Effect of Spatial Organisation in Response Threshold Models for Social Insects. In *Proceedings of the European Conference on Complex Systems (ECCS '09)*. University of Warwick, UK.

Diwold K, Merkle D, and Middendorf M (2009a). Adapting to Dynamic Environments: Polyethism in Response Threshold Models for Social Insects. In *Advances in Complex Systems*, 12(3): 327–346.

CURRICULUM VITAE

Zur Person

Name: Konrad Diwold
Anschrift: Talstr. 34, 04103 Leipzig
Geboren: 7. März 1982 in Wien, Österreich
Nationalität: Österreich

Wissenschaftlicher Werdegang

2000 Matura (Abitur) am BRG 18 Schopenhauerstrasse in Wien, Österreich

2001 - 2005 Bachelorstudiengang Cognitive Science an der Universität Osnabrück

Bachelorarbeit: *The influence of dynamic environments on polyethism in response threshold models for social insects*; Betreuer: Prof. Dr. K.–U. Kühnberger (Universität Osnabrück), Prof. Dr. M. Middendorf (Universität Leipzig)

2005 - 2007 Masterstudiengang Artificial Intelligence an der Vrijen Universiteit Amsterdam, Niederlande
Spezialisierung: Organisational Dynamics and Self-Organisation

Masterarbeit: *Graph-based quantification of self-organisation in molecular simulations*; Betreuer: Prof. Dr. M. Schut (VU Amsterdam, NL), Prof. Dr. A. Troisi (Warwick University, UK)

2008 - today Wissenschaftlicher Mitarbeiter am Lehrstuhl für *Parallelverarbeitung und Komplexe Systeme* an der Universität Leipzig