Fachhochschule
Salzburg University
of Applied Sciences

On the Influence of Information Verification and Propagation on a Swarm's Success

BACHELORARBEIT 2

StudentIn Andrea Distler, 0910601007
BetreuerIn PhD Violeta Damjanovic

Salzburg, 21. Mai 2012

Eidesstattliche Erklärung

Ich erkläre hiermit eidesstattlich, dass ich die vorliegende Bachelorarbeit selbständig und ohne fremde Hilfe verfasst, und keine anderen als die angegeben Quellen und Hilfsmittel benutzt habe. Weiters versichere ich hiermit, dass ich die den benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungskommission weder im In- noch im Ausland vorgelegt und auch nicht veröffentlicht.

Datum Unterschrift

Kurzfassung

Vor- und Zuname:	Andrea DISTLER
Institution:	FH Salzburg
Studiengang:	Bachelor MultiMediaTechnology
Titel der Bachelorarbeit:	On the Influence of Information Verification and Propagation on a Swarm's Success
Begutachter:	PhD Violeta Damjanovic

Für Schwärme gibt es viele mögliche Anwendungsgebiete in der modernen Technologie. Sie können in Bereichen von tauchfähigen Robotern bis hin zum dynamischen Management vom Stromnetz verwendet werden. Die Kommunikation ist ein wichtiger Teil eines Schwarms, welcher ihn von einem simplen Partikelsystem unterscheidbar macht und das Verhalten des Schwarms mitbestimmt.

Agenten haben festgelegte Ziele in einem Schwarm. In dieser These wird der Einfluss vom Sammeln der Information, der Verifizierung und der Verbreitung derselben auf den Schwarm-weiten Erfolg analysiert. Ein Schwarm mit definierten Fähigkeiten wird unter drei verschiedenen Kommunikationseinstellungen getestet. Der Fokus der These liegt auf dem Einfluss der Bewältigung von Information auf die Leistung eines Schwarms in Hinsicht auf die Erfüllung seiner Ziele.

Schlagworte: Kommunikation, Schwarmintelligenz, Verifizierung von Information

Abstract

Swarms have many applications in modern technology. They have reached fields of applications from submersible robots to the dynamical management of the power grid. Its communication is an important part of a swarm which makes it discernible from simple particle systems and defines much of its behavior.

Agents of a swarm have certain goals. The influence of information gathering, verification whether information is still up to date and propagation in the swarm on the overall swarm success are analyzed in this thesis. A swarm with defined skills is measured with three different communication settings. The focus of the thesis is on the influence of information handling on a swarm's capability of fulfilling goals.

Keywords: *Communication, Swarm Intelligence, Information Verification*

Contents

List of Abbreviations

DDOR Distributed Dynamical Omnicast Routing

1 Introduction

Systems of communicating agents have long been part of the software industry. Swarm dynamics are used in network communication, robotics, sensor fields, simulations, and video games. Dynamic swarms are described as collections of elements “belonging to the same category.” (Ghanem and Ahuja 2011) These swarm elements “occur in a dense spatial configuration”, their “motions are statistically similar”, and “the motions of the swarm elements are globally independent.” (Ghanem and Ahuja 2011) Important tasks of swarms are often exploration of unknown and/or changing environments with certain points of interest, clustering of information and way more.

In network communication, swarms can dynamically choose routes between servers. In (Vytelingum et al. 2010) an application called Smart-Grid for swarms in network energy distribution is described.

In robotics, swarms can be used for a multitude of applications. One solution to the problem of surveillance of big areas is described in (Li et al. 2006). Generally, robot swarms can be seen as sensor networks, because most robots have some kind of sensors to be able to gain information from their surroundings.

In video games, swarms are usually in charge of the movement of vehicle groups or other mobile agents. For example, there are games in which neural networks with static agents have been used.

Chapter 2 presents related work in the above mentioned main fields of swarm-based implementation techniques.

1.1 Motivation

Communication is an essential part of swarm intelligence.

> “ “If you and I swap a dollar, you and I still each have a dollar. If you and I swap an idea, you and I have two ideas each.” By openly sharing ideas and work, a team’s creative output is exponentially more than the sum of creative outputs of all the individual team members. While swarm intelligence is based on equal sharing of information, swarm creativity is founded on sharing ideas openly.”(Gloor 2006, p. 22)

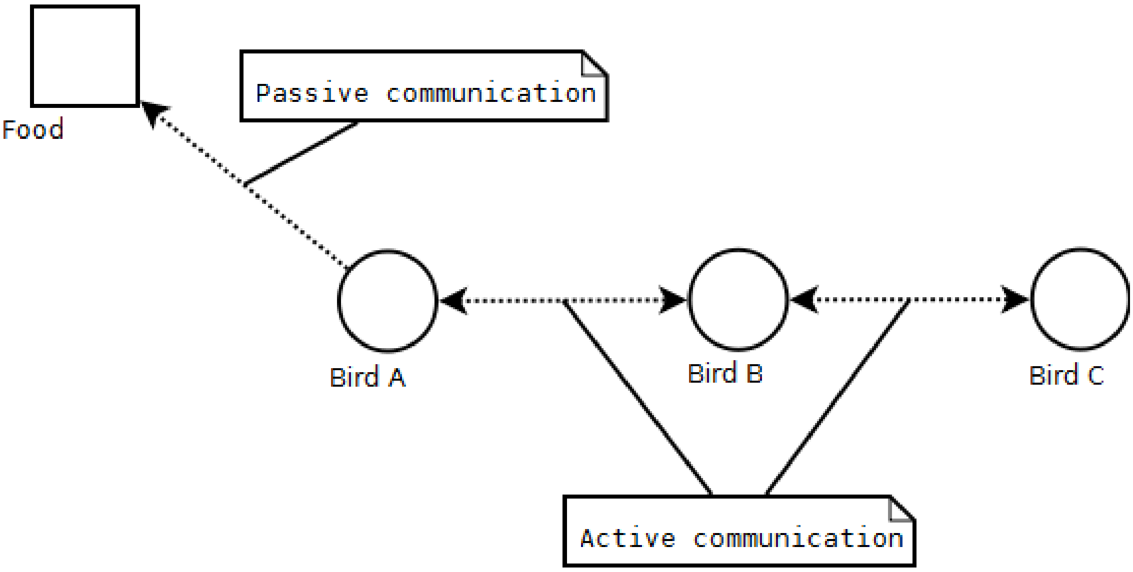
Therefore the means of communication is the most influencing factor in a swarm. If the communication is badly designed, the swarm will not be able to distribute information as well as when communication happens smoothly. The communication between agents defines the swarm, but it is difficult to create metrics for good and bad communication.

This thesis describes three different ways of communication (see chapter 4) and rates them with regard to the swarms’ goals (see chapter 5). The swarms that best reach their goals with the best rating are the systems with the better communication.

Communication in swarms can be categorized as passive or active communication. In figure 1 both forms of communication are visualized.

Passive communication is a one-way communication type. For example, in flocks of birds passive communication would include keeping the distance to other birds. Bird A observes bird B. Bird B offers information, which can be gained without any active involvement of B, like position and velocity. Bird A can gain this information simply by looking bird B and then use this information to avoid colliding with B. Gathering information from food is also passive.

Figure 1: Gathering information from food is a one-way form of communication and therefore passive. Exchanging information with other birds is a two-way form of communication and thus active.



Active communication is a two-way communication type. This behavior can be observed for example in humans. One person says something, the other reacts and says something else. When defining active communication like this, it becomes clear that on a shorter time span active communication is nothing more than passive communication. It is an episode of multiple one way communication efforts, where the single efforts are connected by context.

The main goal of a swarm-member is to satisfy its needs. If many swarm-members succeed in reaching this goal, a swarm succeeds in the swarm-goal of satisfying the needs of as many agents as possible.

In the swarm simulation used in this thesis, the swarm members' needs are called *information* and *hunger*. Depending on how well each of these two are satisfied, the swarm members take individual action. The connections between needs and actions are shown in table 1.

	hungry	not hungry
uninformed	gather information (communicate and explore)	gather information (communicate and explore)
informed	eat	communicate

Table 1: Overview of needs and reactions of agents. Gathering information describes both active and passive communication. Exploring is categorized as passive communication because there is no exchange of information between the food and the swarm-member.

For details of the implementation refer to chapter 4.

1.2 Research Question

How can information propagation influence the success of a system of communicating agents in a swarm in reaching specified goals?

The research presented in this paper is executed in the field of swarm intelligence. It is a qualitative study which determines the links between the communication in a swarm and its success. This is done by implementing a swarm which has a defined way of moving about. Its agents have defined needs and a defined way of communication. The communication between swarm members can be configured easily. The three different ways of communication mentioned above are created by configuring the swarm's communication in three different ways. They can be described as follows:

1. The first way of communication is configured so that the swarm members can only "see" other members and not "talk" to them. There is no active communication and no actively passing on of information between swarm members. This means that swarm members can only react to the movement of other members.
2. The second way of communication defines the swarm such that members "talk" to others but deem even very uncertain information as significant. These members will estimatedly need longer than the ones from the third configuration because they pass on more information.
3. The third way of communication makes swarm members dismiss uncertain information and only pass on very certain information. These members are fast "talkers" as they do not pass on as much information. However, they could be dismissing information which is still accurate despite being rated as uncertain.

The success of a swarm is evaluated by measuring how many members are well informed and well fed. It can only be determined which of the three configurations enables a swarm to survive better *in comparison to the others.*

1.3 Structure

Chapter 2 presents the state of the art techniques in swarm intelligence. In chapter 3 the expected outcome of the implementation is discussed. In chapter 4 the implementation of the swarms and the three ways of communication are described. In chapter 5 the means of measuring the outcome are described and the results gained from the implementation are presented, analyzed and compared to the expected outcome. Finally, chapter 6 gives the conclusion remarks.

2 Related Work

2.1 Movement

2.1.1 Decentralized movement

In *Flocks, herds, and schools: a distributed behavioral model*, Reynolds presents an alternative way to modelling bird flock, fish schools or similar movement by scripting each swarm member's path individually. Reynolds' approach includes giving each swarm member a mind of its own and letting the swarm members based on a set of rules decide where they should be in the next timestep of the simulation.

Reynolds derives his swarm from particle systems where each particle has abilities and properties in addition to those of ordinary particles. This enables the simulation of natural flocking movement. Particle systems are very similar to flocks, because they "are collections of large numbers of individual particles, each having its own behavior" (Reynolds 1987). One important difference between particle systems and swarms described by Reynolds was the missing interaction between particles. Lately, this boundary has started to blur, as fluid and similar physics simulations require a great deal of interaction between particles.

> "The basic urge to join a flock seems to be the result of evolutionary pressure from several factors: protection from predators, statistically improving survival of the (shared) gene pool from attacks from predators, profiting from a larger effective search pattern in the quest for food, and advantages for social and mating activities [33]." (Reynolds 1987)

All of these tasks seem to rely heavily on the swarm's motion. The three famous rules defined by Reynolds are:

1. Collision Avoidance: avoid collisions with nearby flockmates
2. Velocity Matching: attempt to match velocity with nearby flockmates
3. Flock Centering: attempt to stay close to nearby flockmates

The motion of a swarm is not determined by one single agent or any centralized unit, it is created by the sum of movements of the single agents. A flock member does not need to know about all of the other flock members. As Reynolds describes, a bird might be aware of three categories: itself, its two or three nearest neighbors, and the rest of the flock. Flock members use a constant time algorithm for calculations of their movement as a part of the flock, which is why flocks are indefinitely scalable in nature. Reynold's model does not directly simulate the limited senses of a flock member, but filters the globally available information for members such that they have approximately the same amount of information as natural flock members may have, therefore information in the flock is strongly localized. This leads to the typical behavior of flock members.

2.1.2 Force Field Animation System

Reynolds mentions another approach for dynamic simulation of flock movement which is called "the force field animation system". In this model particles follow a force field and dynamically change the forces around themselves such that particles seem to avoid collisions. In the same way particles avoid collisions with static objects. The animator sets the initial values for velocity, orientation etc. The rest of the simulation happens automatically.

2.1.3 Relevance

In this thesis, a mixture of both approaches is used to animate the movement of flock members. The exact implementation is explained in chapter 4.

In the papers from Reynolds and in the force field animation system, the focus is on a swarm's movement and not on the communication of information. The only communication happening is indirect by averaging the movement of surrounding members. The only communication used in the second approach serves to keep the distance between particles.

2.2 Communication

2.2.1 Labor Allocation with Response Thresholds

In (Bonabeau, Dorigo, and Theraulaz 1999) the mechanics used by ants and other insects for fulfilling several tasks are explained. These tasks include finding the shortest path, foraging, cemetery organization, brood sorting, and clustering. The authors explain how these tasks can efficiently be allocated to different individuals of a group of coworking insects in nature and simulations. It is of special interest, because labor allocation implies some kind of coordination, which in turn requires communication between group members. The allocation of labor will be explained in detail here, but the detailed description of the tasks would go beyond the scope of this thesis.

> "In social insects, different activities are often performed simultaneously by specialized individuals. This phenomenon is called division of labor."

(Bonabeau, Dorigo, and Theraulaz 1999, p. 110)

Social insects which are allocating tasks tend to form *castes*. Members of a caste perform similar tasks. The authors pose three categories of labor allocation:

> 1. *Temporal polyethism.* With temporal polyethism, individuals of the same age tend to perform identical sets of tasks. Individuals in the same age class form an *age caste.* [...]
> 2. *Worker polymorphism.* In species that exhibit worker polymorphism, workers have different morphologies. Workers that differ by their morphologies are said to belong to different *morphological or physical castes.* [...] An example of a worker caste is the soldier or major caste which is observed in several species of ants.
> 3. *Individual variability.* Even within an age or morphological caste, differences among individuals in the frequency and sequence of task performance may exist. One speaks of *behavioral castes* to describe groups of individuals that perform the same set of tasks within a given period.

Labor allocation is generally an ongoing process which dynamically adapts in response to "internal perturbations or external challenges". Important influences are food, risks and danger, weather, time of the year, and colony development.

Bonabeau, Dorigo, and Theraulaz present a model to simulate task allocation.

> "In this model, every individual has a response threshold for every task. Individuals engage in task performance when the level of the task-associated stimuli exceeds their thresholds." (Bonabeau, Dorigo, and Theraulaz 1999)

When individuals with low response thresholds for a specific task are removed, the need to get their former tasks done grows and individuals with higher response thresholds take over their places.

> "The nature of task-related stimuli may vary greatly from one task to another and so can information sampling techniques, which may involve direct interactions among workers [...] or more or less random exposure to task-related stimuli." (Bonabeau, Dorigo, and Theraulaz 1999)

This response threshold model can be extended by changing the response thresholds when insects incur or do not incur certain tasks. By reinforcement learning insects' thresholds can be adapted such that individuals which have performed a task will more likely perform the same task again.

For simplicity, a fixed task threshold model is used for the approach presented in this paper. The need for food is abstracted by a value. It is zero when the agent is not hungry and it increases over time. Once the need reaches a certain threshold its goal changes to eating.

2.2.2 Omnicast Communication by Submersibles

In (Schill and Zimmer 2006) the communication in schools of submersibles is discussed and an approach called *Distributed Dynamical Omnicast Routing* is tested in practical applications.

Under water information transmission via wireless channels has limited bandwidth and range. The submersibles have a limited sensor range and can only deliver sparse information about the underwater environment.

> "It is crucial that submersibles share their local information with their direct neighborhood (local control information, positions, etc.) and the whole school (measured gradients, maxima and minima of sampled data, global control parameters, etc.). The most suitable form of communication is [...] *omnicast*." (Schill and Zimmer 2006)

Omnicast is a form of many-to-many communication and is also known as *gossiping*.

The communication network is modelled as a graph, where vertices represent swarm members and edges are bi-directional, non-weighted, and represent a communication link. A node n_i can receive exactly one message. If two nodes send a message at the very same time, neither of the messages can be received. This is called a collision in node n_i. The problem with the discussed half-duplex omnicast communication is to create an information exchange schedule which propagates information as fast as possible and avoids collisions. The optimal communication algorithm would ensure that all nodes receive all information in minimal time.

> "The information which is available to each node is only the information contained in messages which they receive. In practice this means that nodes can become aware of their 2-hop neighbourhood." (Schill and Zimmer 2007)

The definitions above assume a "static case in which all members synchronize before a new set of information tokens becomes available." (Schill and Zimmer 2006) For practical use the synchronization phase is omitted. The individual omnicast schedules overlap.

The *Distributed Dynamical Omnicast Routing* method has been measured and found to be performing "well with regard to information dispersal both in simulations and reality."

2.2.3 Virtual Power Plants of the Smart Grid

In (Dimeas and Hatziargyriou 2007) the focus lies on controlling virtual power plants of the Smart Grid with social agents. A virtual power plant is a kind of alliance of lower-level production units which “cooperate in order to participate as a single entity in an energy or CO_2 market.” Virtual Power Plants are explained more detailed further below. The paper proposes the use of multi-agent system technology to improve existing approaches and shows by example “how local intelligence and communication can be used in order to achieve the related complex tasks.”

> “An agent is capable of acting in the environment, meaning that the agent is capable to change its environment by its actions. For example, an agent that controls a storage unit and decides to store energy, rather than to inject it, alters the decision and the behavior of other agents.” (Dimeas and Hatziargyriou 2007)

Agents can also communicate with each other and with the Market Operator. In (Vytelingum et al. 2010) it is mentioned that to predict future power and storage usage, agents can procure weather and similar information. Agents have a certain authority and can decide on their own without asking a hierarchically higher central unit about this specific decision. This helps keeping the communication network and infrastructure small which in turn reduces the possible communication strain on the Smart Grid.

Agents have “partial or none at all representation of the environment.” (Dimeas and Hatziargyriou 2007) Each agent knows only about its own state, but it can obtain information about other agents or the environmental system via communication. In addition to communication, agents have a certain behavior which is determined by their goals, their current states, and their skills. Agents can also offer services to the system. For example an agent can sell power on the market, which is seen as a service. “Depending on the goals of the unit owner, the various units in a Microgrid can behave mostly autonomously in a cooperative or competitive environment.” In case an agent can not operate any more, other agents can fill in for it. This adds a “plug-and-play” capability to the system which greatly enhances its reliability.

In the paper Dimeas and Hatziargyriou show an example where 170 households, which cannot participate in the market independently, form a multi-agent system to be able to take part in the market.

In order to control the system the participating entities need to be categorized in three hierarchical levels. These levels are visualized in figure 2.

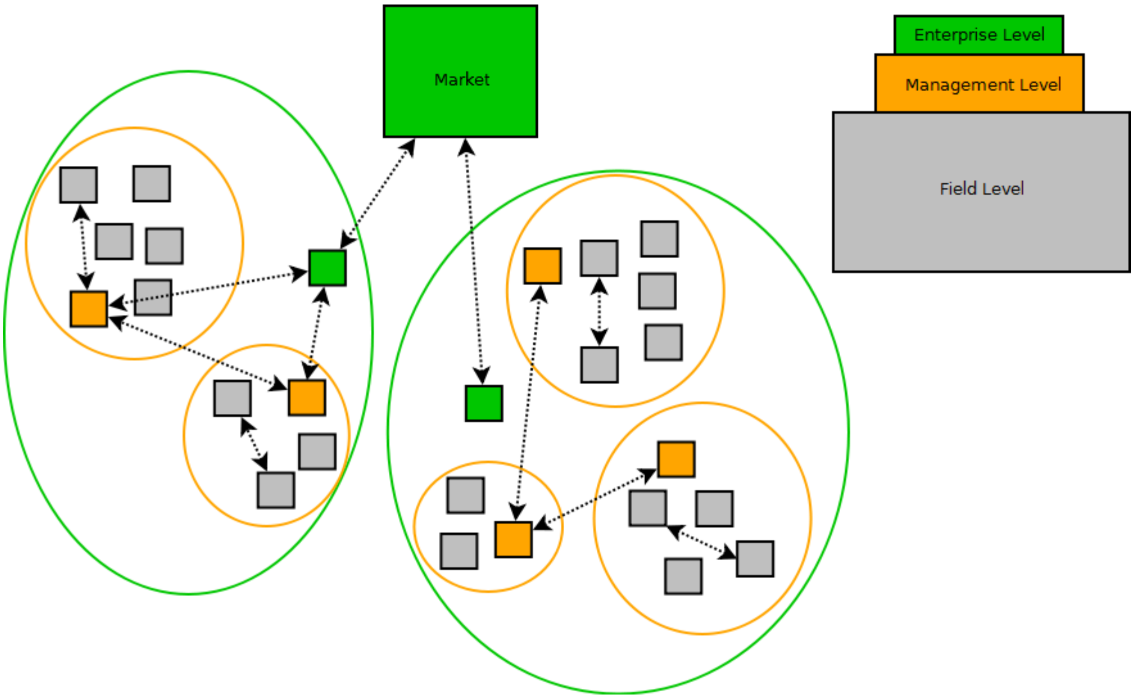
The *field level* is the base of the system. It consists of “all agents associated with the control of the production units or controllable loads”. The agents belonging to the field level communicate directly and control production units.

The second and middle level is the *management level*. Each multi-agent system (for example the 170 mentioned households) has an agent which is responsible for communication with other similar multi-agent systems with the desire to cooperate. These middle agents belong to the management level.

The small multi-agent systems can ally and form larger multi-agent systems to be able to participate in the energy market at the *enterprise level*. This definition of layers makes it possible to create nested layers, because each multi-agent system needs a representative, which communicates with other multi-agent systems’ representatives. This strategy blurs the borders between the layers. However, they can still be determined by the number of agents in a multi-agent system.

The authority an agent has and therefore the decisions an agent is allowed to make depend on the agent’s level. Agents on level one can be responsible for battery management and are allowed to make decisions “that are necessary to control and manage locally the unit.” Decisions on level two influence agents of level one. Level two decision are typically whether to store energy or to feed energy into the system. They are carried out by agents of level one. Complex decisions are made

Figure 2: Devices on the field level ally to form bigger groups, which can participate on the management level. One of the field level devices is responsible for communication with other management level devices. These multi-agent system on the management level can work with other multi-agent systems to be able to participate in the energy market at the enterprise level.



on the enterprise level. For those decisions, the state of the system is very important, however it is highly likely to be partially unkown. The outcome of level three decisions can be unkown and it can have long term effects. An example for a level three decision is whether to buy or sell energy from or to the grid.

3 Hypothesis

3.1 Group Intelligence

The research problem discussed in this paper, is how different information propagation can influence the success of a swarm (see chapter 1.2). However, this is not a testable statement and needs to be refined, which is done in the following chapter.

It is an accepted fact that groups of intelligent elements exhibit behavior far beyond the sum of capabilities of all single entities of the group.

> "An insect may have only a few hundred brain cells, but insect organizations are capable of architectural marvels, elaborate communication systems, and terrific resistance to the threats of nature." (Kennedy and Eberhart 2001, p. 99)

One of the things differing a group of individuals from a swarm or flock is that the individuals of the latter communicate. The assumption that the communication in a swarm affects the swarm's success is not a farfetched one. However, it is not clear to what extent the communication influences the swarm's success.

The center of the hypothesis is the influence of the swarm's communication on the swarm's success: *A swarm's communication can influence the swarm's success.* With the purpose of creating a verifiable or falsifiable statement, the communication and the success of the swarm need to be defined.

3.2 Information Propagation

Agents move in an environment which offers information under certain circumstances. If an agent complies with the requirements, it may gather information. For an agent, the other agents of the swarm are part of the environment. Information can be gathered from fellow agents by complying with the information access requirements. Once an agent retrieves a piece of information, it can decide what to do with it and, in the case of this paper, whether to remember or discard it.

A bird in a flock of birds in nature communicates via watching their neighbors and averaging their movement. Generally speaking, bird A obtains information and bird B obtains it from bird A. This can be repeated infinitely. Since each bird averages the information it obtains to a certain degree, the propagated information is changed and never exactly the same in bird B as in bird A. This system forms a cycle of constant information propagation and averaging, because bird A can re-retrieve the information which bird B has retrieved from bird A and changed.

Information propagation is subject to limitations. Examples for these are proximity, communication exchange duration, bandwidth, and many more. These limitations can dictate whether information is exchanged at all, how long the exchange takes, and whether the information is distorted when it is exchanged.

Therefore it can be stated that the *way of information propagation between members of a group influences the overall success of this group.*

3.3 Information Categories

In this thesis three ways of information propagation are compared. A first overview of those three different communication settings was given in chapter 1.1. They are discussed in more detail here.

There are two categories of communication: active and passive communication. Passive communication is similar to the form of communication described in (Reynolds 1987) in which the exchanged information contains information about the neighboring birds' movement and orientation. It is a one-way kind of information exchange and does not include actively giving information to others. The information which is "exchanged" is openly on display and can be copied anytime. Active communication is similar to a conversation between a number of people. Partner A gives information to partner B, partner B receives it and processes it as wished. In a conversation it is possible for partner A to ask partner B for information and partner B can decide whether to tell partner A something or not. For active communication it is imperative that both partners are open for the communication.

For example, suppose a conversation that goes like this:

1. Partner A asks partner B for information.
2. B gives information to A.
3. A gives B some other information.

This conversation would clearly be an active propagation of information, for both A and B. If partner B just would not answer in the beginning, no information would be exchanged and no conversation could be established. If partner A could not receive the information sent by B, the same would happen.

From the three forms of communication introduced in chapter 1.1, two can be classified as active communication and one as passive only.

3.4 Agents' Communication Settings

The passive communication setting does not allow members of the flock to pass on information actively. The only information which can be retrieved is information which is freely displayed and which does not require establishing a two-way communication link. This is similar to the approach originally described in (Reynolds 1987).

There are two communication settings where members exchange information actively. When information is passed on, it becomes more uncertain with age and different storage or averaging in each flock member. The flock members can decide what to do with information. The difference between the two communication settings categorized as active is a measure of information importance called *certainty threshold*. One kind of flock has a low certainty threshold, and the other has a high certainty threshold. Information can be rated from uncertain to certain. If a piece of information is certain enough to remember, it will be stored in an agent's memory and can be spread to other agents.

Based on the theory that communication has a strong influence on the success of a swarm, these three different settings should bring very different outcomes and therefore support the theory. The interesting part of the hypothesis is the question *how* communication can influence the success of a swarm. This question is analyzed by comparing the outcomes and the success of three swarms which have the same means of movement and communication, but different communication settings.

The first group of agents has got no means of communicating directly. Its agents' certainty thresholds are set such that they only deem information significant if they have gathered it themselves. That makes communication impossible, because no other agent can offer information of significance. This does not include passive communication. All three flocks' agents can act on what they see.

The second group of agents has got the means of communicating directly. These agents only remember information which is very significant. Uncertain and therefore possibly wrong information is ignored. This gives them the advantage of only acting on certain information. However, it cannot be guaranteed that the information is true.

The third group of agents has also got the means of direct, active communication. Their certainty threshold is very low, which estimatedly results in much wrong information circulating even after the original source of information is gone. This also defies the disadvantage of discarding perfectly valid information of a high certainty threshold.

The estimated advantages and disadvantages of the three introduced ways of communication are presented in table 2.

Table 2: Probable advantages and disadvantages of each communication model.

	advantages	disadvantages
no active communication	agents only handle information which is most certainly true, do not squander time on “talking” and have more time to explore	the gathered information is restricted to the agent which has gathered it
high certainty threshold	agents only handle information which is possibly true, do not need as much time to exchange information as with a low certainty threshold	information which is still valid may be discarded
low certainty threshold	agents do not discard information which could still be true, generally higher information circulation	invalid information is not discarded but treated as though it was accurate, leading to wrong behavior

These three different settings of the certainty threshold should bring forth distinct behavior in the agents.

The hypothesis so far is: *Three different certainty thresholds exhibit distinctly different behavior in the agents of a swarm and therefore directly influence the overall success of the swarm.*

In the swarm which does not actively communicate, the spread of information does not happen as fast as in the other two swarms, because it misses the important means of propagating information. Thus the agents without communication cannot consider as much information as the agents of the other two swarms when making decisions about their behavior. It is expected that agents of this swarm are generally less informed and that these agents perform poorly because of that low information rate. More information creates a better base for deciding and for exhibiting adequate behavior.

Agents that have a high certainty threshold do not remember as much information as agents which have a low one. However, high certainty thresholds assure that remembered and passed on information is more accurate. This means that behavior based on this information is also more accurately fitted to the situation. The time needed to exchange information is less compared to the agents with a low certainty threshold because there is less information to exchange, assuming that the time needed to exchange one piece of information is constant and the same for each piece of information. Agents that have a low certainty threshold remember information which would be discarded with a higher certainty threshold. Therefore they have a wider range of information to base decisions on. A downside to this is that there may be information remembered which is outdated or inaccurate in another way. This approach creates a higher tolerance for wrong information, but also makes it possible for accurate but rated as uncertain information to be included in the decision-making. Another undesirable effect created by a low certainty threshold is, that exchanging all information which has been retrieved by two agents needs more time and therefore there will be less time to gather new information from the environment.

The differences between the high certainty and the low certainty communication settings have been discussed above. It is up to the evaluation chapter to describe what the observed differences are and to compare them to the assumptions made in this chapter.

3.5 A Swarm's Success

The success of the swarm can be any goal which can be reached by exhibiting a certain sequence of behavior. In this thesis, the goal for an agent is to not be hungry. This can be achieved by firstly, gathering information about food, then secondly, going to the food and then thirdly, eating it. The secondary goal of an agent is to spread information. This goal is pursued only when the first is not pursued. This situation occurs right after an agent has eaten.

3.6 Hypothesis and Null Hypothesis

The behavior of agents is defined by these two goals and by the information they are able to gather. Therefore the final hypothesis is: *Different certainty thresholds influence the way agents cope with information and thus the sequence of behaviors exhibited by agents intent on reaching specified goals.*

The null hypothesis is: *Different certainty thresholds do not influence the sequence of behaviors exhibited by agents.* If the outcome of the experiment fits the null hypothesis, the hypothesis is falsified.

4 Implementation

4.1 Overview

This chapter describes the experiment's design and its components. The three different flocks and their different behavior is explained in detail. Finally, the method used for gathering of data is reported.

This thesis follows an experimental approach to either verify or falsify the hypothesis. Three flocks of artificial bird-like agents are implemented as similarly as possible. The unavoidable differences in their implementation are described in section 4.5. The independent variable in this research is the certainty threshold which is different in all three swarms. The influence of this variable on the swarm's behavior and specifically on its success is measured.

Agents move in an environment. It is described in 4.3. The environment has elements which can be found by agents when they explore the environment. The elements offering information in this implementation are called food and will be described in section 4.4. When an agent finds one of these elements, it can gather the information this element offers. When an agent's primary needs are satisfied, it starts conversations with other agents more willingly.

4.2 Used Technology

4.2.1 Unity Game Engine

The implementation was made in the Unity Game Engine[1]. It is a proprietary Game Engine which also includes an editor. This makes it possible to create basic prototypes of an application in a very short time. The game engine can be used with different programming languages. For the implementation discussed in this chapter, C# was chosen as programming language. The engine uses Nvidia PhysX[2] as physics framework. Unity can deploy applications on many platforms. The development operating systems are Windows and Mac OS X.

One reason why the author has chosen the Unity Game Engine is because of the elaborate trigger system. The agents' senses are implemented using these triggers. The triggers can be adjusted using the engine's user interface. Another advantage of the Unity Engine is that the visualization of agents, their state changes, and the information flow between them is very easy and can be rapidly implemented.

4.2.2 Visual Studio 2010 and MonoDevelop for Unity

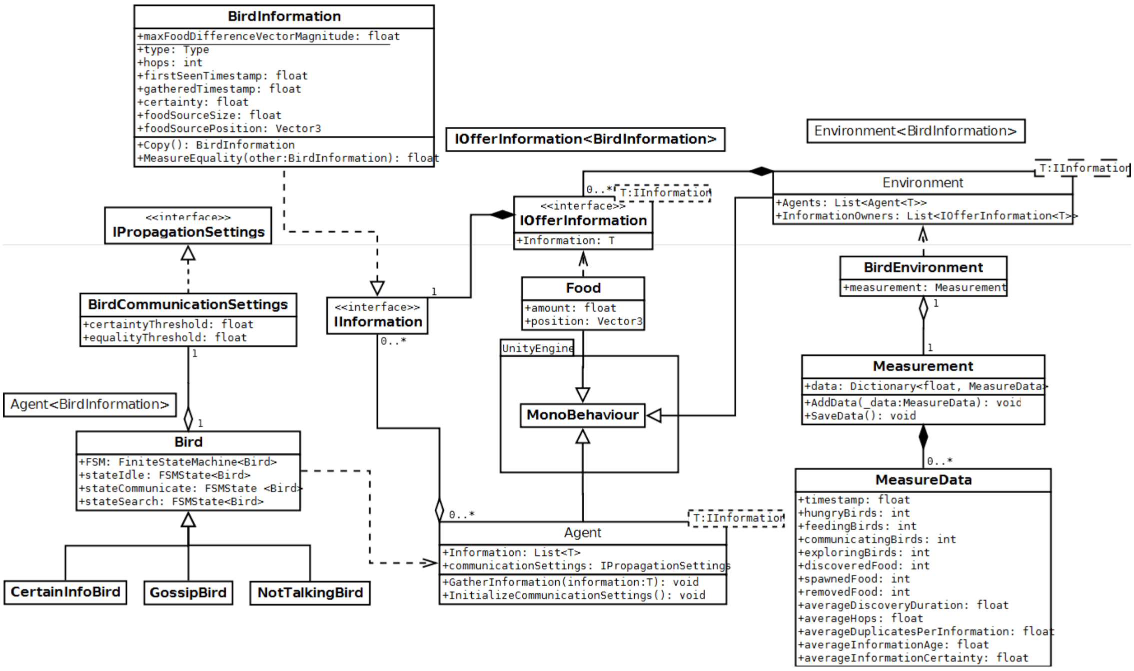
Visual Studio 2010 was used to create source code. Visual Studio is a integrated development environment produced by Microsoft specifically for deployment on the Windows operating system. It supports the creation of a variety of applications. Using Unity and Visual Studio together creates problems when trying to debug an application. A special version of MonoDevelop, which was created to work together with Unity was used for debugging. MonoDevelop for Unity supports break points and similar features, which do not work with Visual Studio.

4.2.3 Git and GitHub

Git is the version control system used to track different revisions during development. The whole project, including source code and important Unity project files, can be found on gitHub[3].

1. `http://unity3d.com/`
2. `http://www.nvidia.com/object/physx-9.12.0213-driver.html`
3. `https://github.com/JungleJinn/Bakk2`

Figure 3: An overview of the swarm classes.



4.3 Environment

The environment of an agent has the form of a box and specified bounds. The environment manages the creation and handling of food and birds, defines settings, and manages the measurement by bundling all data about agents, food, and information.

The testing environment uses birds as agents. Those artificial birds are approximations of real birds. They are similar in that they both move and eat. Their movement patterns are similar, but only to a certain degree. The artificial birds are described in detail in section 4.5.

The environment's elements are encoded with different colors, depending on their state and whether they are currently exchanging information.

4.4 Food

Food is distributed randomly in the environment. The amount of food created in the environment is tweaked by the user. It can be exactly as much as all birds can hold, less, or more. There may be differences in performance of the flocks when there is more food than they can eat or less than that. Detailed information can be found in chapters 4.7 and 5.1. When a food source is depleted, a new one appears at a random position with a random amount of food.

Food offers information to a bird once it is discovered. Food can be discovered by a bird when it has moved close enough such that the food is in the bird's *visibility trigger*. When a bird discovers food, it gathers the offered information. This action is indicated by a flash of white color on the surface of both elements. Food can hold exactly one piece of information. This piece of information stores data like the amount of food this source can offer and the source's position.

Food has got two states: *undiscovered* and *discovered*. Once a bird has seen the food and gathered its information, the food instance is discovered. These two states are color-encoded. When food is discovered, it is colored violet, else it is colored green. This helps to quickly get an overview of how well the environment is surveyed by the birds.

The amount of food a food source currently holds is expressed by its size. A food source with much food is bigger than one with little food. This makes it slightly less likely for birds to discover a small food source, because finding elements depends on distances. There have not been visible effects, most likely because the environment's small size equilibrates that.

The relationships between food, information, environment and birds are described in figure 3.

4.5 Agents and Birds

Agents are an abstract class which defines the most important attributes and methods of an agent acting as a swarm member. These include:

- The type of information which is to be used.
- A list in which to store the gathered information.
- The communication settings.
- The method used when gathering information.
- The method used to initialize the communication settings.

The bird class implements the functionality of the birds. There are three classes deriving from the bird class. These are called *NotTalkingBird*, *CertainInfoBird* and *GossipBird*. They only override the method used to initialize the communication settings. All of the other functions are exactly the same, except with the NotTalkingBird which implements a switch to turn off the communication with other birds of the swarm, so that they cannot change into the communication state. Details to the different kinds of birds can be found in section 4.5.3.

The abstract agent class is implemented in the code listing 1

Listing 1: The agent class, defining the basic structure of birds.

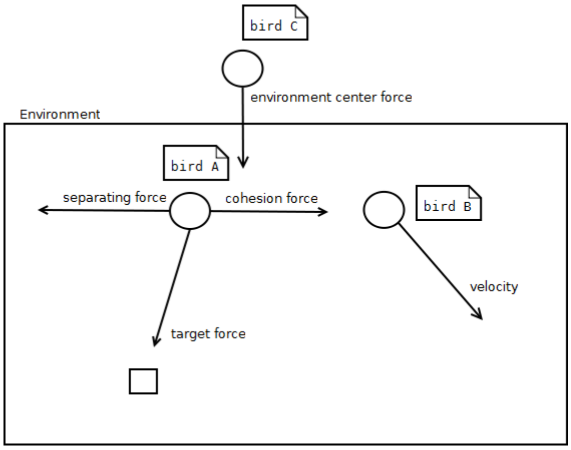
```
// Agent.cs

using UnityEngine;
using System.Collections;
using System.Collections.Generic;

/// <summary>
/// An agent has certain settings for information propagation. It owns
/// a list of all the information it has already gathered.
/// </summary>
/// <typeparam name="T">The type of information which can be gathered
    by this agent</typeparam>
public abstract class Agent<T> : MonoBehaviour where T : IInformation
{
    /// <summary>
    /// A list holding all the information an agent has gathered.
    /// </summary>
    public List<T> Information;

    /// <summary>
    /// the most important settings in this project: the
       communication settings.
    /// They can be different for each bird (however, are only
       different for the three swarms)
    /// </summary>
    public IPropagationSettings communicationSettings;

    /// <summary>
    /// This method determines how information is gathered and
       whether information is gathered at all.
    /// </summary>
    /// <param name="information"></param>
    public abstract void GatherInformation(T information);

    /// <summary>
    /// Initializes this agent's communication settings
    /// </summary>
    protected abstract void InitializeCommunicationSettings();
}
```

4.5.1 Movement

The movement of birds is implemented using a mixture of both approaches described in chapter 2.1. A bird is only influenced by forces. These forces are applied in the global state of each bird. Parallel to the global state, the specified states are updated. Each specified state sets multipliers, which are used to control the magnitude of certain forces. Details on the states can be found in the next section (chapter 4.5.2).

Reynolds describes in his paper “Flocks, Herds and Schools” the movement of birds which depends strongly on neighboring birds. Following his approach, there is a force which is called *cohesion force*. It keeps birds close together. Another force is called *separating force* and it keeps birds

Figure 4: Bird A is influenced by three forces. The separation force keeps birds separated, the cohesion force keeps them together, and the target force points towards a defined target. The direction of A is further influenced by neighboring birds such that their velocities are averaged. Bird C experiences a force which leads it back into the environment.



separated. There is another force for keeping birds inside the environment. This force is based on the force field method mentioned by Reynolds. Additionally, there is a force, which aids a bird in reaching a target point. The different forces are visualized in figure 4.

Additional to the forces mentioned above, there is the averaging of neighboring velocities, which creates the distinct flock movement patterns. In this thesis, these movement patterns have been modified to fit the task of surveying the environment better.

Collision avoidance techniques have not been implemented for this experiment.

4.5.2 State Changes

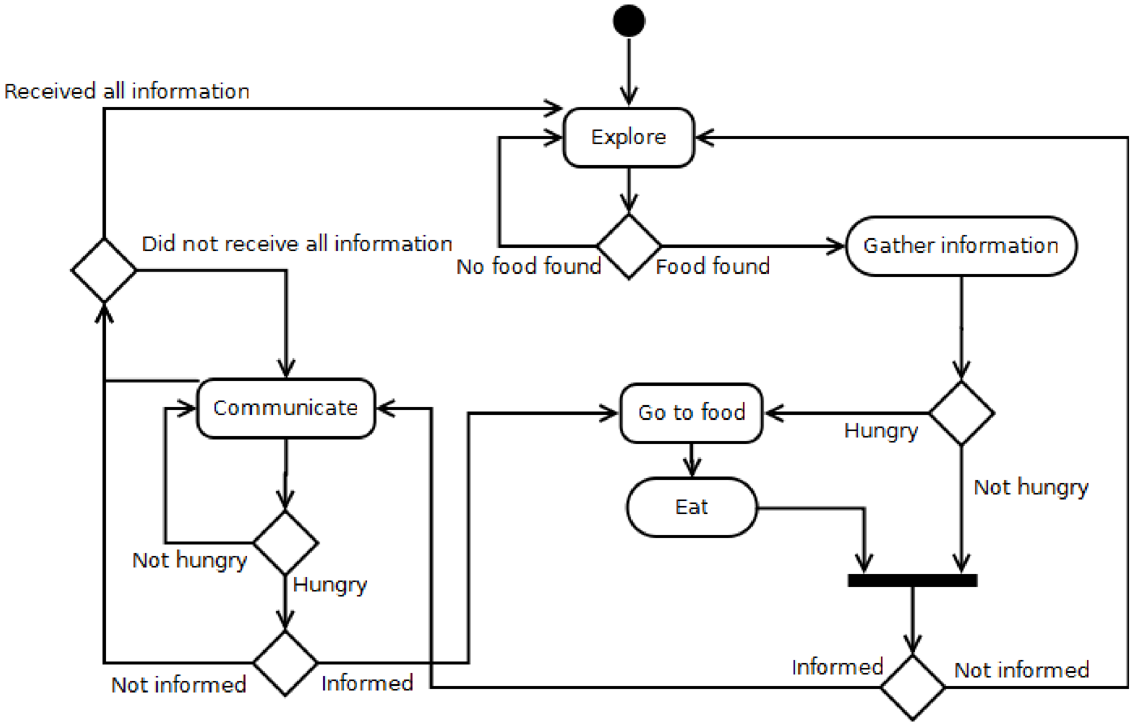
A bird can be in one global state and three specialized states. The global state handles the integration of the forces mentioned above into the bird's movement and the forces are calculated in the global state. The multipliers used to influence the forces' magnitudes are set in the specialized states.

The specialized states are:

- The *eat* state. The nearest source of food is chosen and set as the new target point. The target point force multiplier is distinctly bigger than the other multipliers. If the target point has been reached but there is no food, the information is deleted.
- The *exploration* state. When birds are in this state, they strive to keep more distance between themselves and other birds than usual. The goal of this state is to discover food.
- The *communication* state. This state is entered when a bird meets another bird and both birds want to exchange information. It has a very strong cohesion multiplier. When birds have much information to exchange, they tend to form clusters. When a bird has no more information to share with the communication partner, it tells its partner. When both do not have any more information of interest to the other bird, they terminate the conversation until they meet again.

The state changes and the most important actions of a bird are visualized in figure 5.

Figure 5: The state change sequences of a bird and its two most important functions: eating and gathering of information.



The first state a bird enters after creation is the explore state. Birds start with an empty stomach and since they do not have any information, they need to get some. Right after the birds have seen that there were about thirty or fifty other birds spawned at the same place, they start talking to each other and form a cluster of communicating birds. When they have discovered that none of them has got any information, they enter the exploration state. There are strong forces repelling birds from each other when they are in the exploration state. This forms an explosion-like movement pattern. The birds start “surveilling” the area. Once they find food, they eat as much as possible. If they are not hungry any more, they start talking to other birds and spread the information they have gathered.

The birds’ state changes are determined by hunger and by “chance”. The need for food is implemented similar to the task allocation thresholds mentioned by Bonabeau, Dorigo, and Theraulaz in “Swarm Intelligence - From Natural to Artificial Systems”. The hunger value rises over time. When it has reached a defined hunger threshold the bird is hungry and acts accordingly. Whether a bird exchanges information with another bird depends on whether it gets close enough to do so. The bird’s movements are different in each simulation because the food is distributed randomly and the food positions have a large impact on the birds’ movements. That’s why the birds’ state changes are also determined by chance.

4.5.3 Communication Settings

The detailed differences in communication are discussed in this section. Each type of bird has different ways of spreading information to other birds. The groups are ordered according to their certainty threshold setting.

Not Talking Bird, Group 1 This type of bird does not have any means of active communication. Its means of information spreading depend on the exploration state, because other birds are mostly ignored in the eat state. When a bird is in the exploration state, strong repelling forces act on it and the other birds of the swarm. This keeps the birds spread across the environment. The birds watch the environment and can react very fast when new food sources appear. If a bird is hungry and discovers a new food source, it will move to it and eat. This changes the repelling forces on its neighboring birds, which move in to fill the empty space. Even though the certainty threshold cannot influence the birds' communication, it still has an effect on information storage. The hop count of some information is always zero, because it cannot be propagated from bird to bird. Thus the hop certainty is always at the maximum. The age of some information still changes. When the information age has reached its pre-defined maximum value (which is very high in this experiment), the certainty of this piece of information reaches zero and will be deleted. The certainty threshold of this type of bird is zero, because in this case the maximum age defines when a piece of information is removed.

Gossip Bird, Group 2 This type of bird can communicate, explore, and eat. Its certainty threshold value is nearly zero. It stores information for longer periods of time and tolerates larger possible error and distortion values from information spreading.

Certain Info Bird, Group 3 This type of bird has the same abilities as the Gossip Bird. It has got a certainty threshold of 0.9. It stores and spreads only very certain information. This value of the certainty threshold is tweaked such that information can only be stored if its hop count is smaller or equal 1.

The possible advantages and disadvantages of the three types of birds are discussed in chapter 3.4.

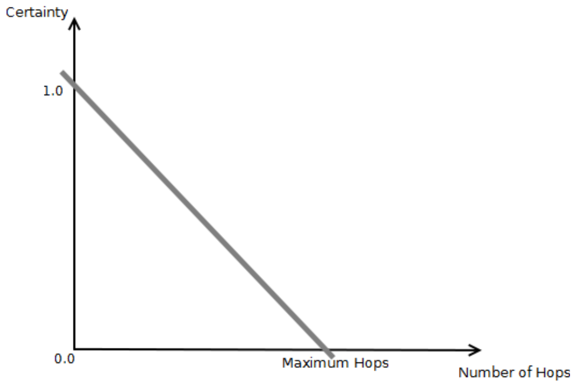
4.6 Information

The information class used by bird agents implements the interface class *IInformation*. It is called *BirdInformation* and it has got several attributes, the important ones are:

- *firstSeenTimestamp*. It stores when this information has first been gathered by an agent.
- *gatheredTimestamp* stores when the owner of this information piece has gathered it. Both timestamps play an important role in the certainty calculations.
- *certainty* needs to be updated by an agent. The certainty calculation is discussed in chapter 4.6.2. Uncertain information is discarded.
- *hops* stores how many birds have possessed this piece of information. When birds share information, the hop count becomes higher.
- *type*. The type of information. In this environment, there is only one type of information, called *food*.
- *foodsourceSize* amount of food the food source offered at the point in time when this information has been gathered.
- *foodsourcePosition*. The position of the food source.

At certain occasions these attributes are filled with information. When a foodsource is distributed by the environment, the type of information, the foodsource's size and its position are updated. The other four attributes can only be updated when the information is gathered by an agent.

Figure 6: The hop certainty function.



4.6.1 Information Gathering

When an agent comes across a new piece of information, it copies the information and creates a duplicate. This can happen in two different situations. The first of these is when an agent discovers a food source it has not already seen. The duplicate created in this situation is not added to the information duplicate count.

4.6.2 Certainty Calculation

The certainty calculation depends on the information's age and its number of hops. The calculation method is presented in listing 2.

Listing 2: The certainty calculation of a piece of information.

```
private float CalculateCertainty(BirdInformation info)
{
        // calculate single certainties
        float hopCertainty = - (float)info.hops /
            (float)settings.maxHops + 1;
        float ageCertainty = - info.age / settings.maxAge + 1;
        // hop and age certainty are equally weighted - calculate
            average of all certainties
        return (hopCertainty + ageCertainty) * 0.5f;
}
```

Both certainty calculations use a linear function of either the number of hops or the age. The function used for the calculation of both certainty variables is visualized in figure 6 by using the example of the hops of a piece of information. The final certainty of the information at a point in time is calculated as an average of hop certainty and age certainty. The information certainty for each collected information is calculated in the update loop of the birds. If the certainty is lower than the certainty threshold, this piece of information is discarded.

There is another possible situation which makes a bird discard information. This happens when a bird is in the eat state and has arrived at the position stored in one piece of information, but there is no food at this position. Then the bird can safely forget that information.

4.6.3 Equality of Information

When birds gather information, they take a look at whether this particular information has already been gathered before. Comparing the information pieces is a challenging task. Information may have been gathered at different times. Also, it cannot be guaranteed that the size of the food source has not changed since the information has been gathered.

This issue has been solved by transforming relevant parts of the information into an attribute space. Each relevant information attribute can be seen as one dimension of the attribute space. After that, the distance of two information vectors is calculated in attribute space and can be used to determine whether two pieces of information are the same or not. When they are close enough to each other, they are essentially the same. The maximum distance vector in attribute space is pre-calculated. The method used to compare information is displayed in listing 3.

The attribute space vectors strongly depend on the type of information, because for each type other attributes are relevant.

Listing 3: The method used to measure the equality of two pieces of information. The distance in attribute space is calculated and the maximum distance is known.

```
public float MeasureEquality(BirdInformation other)
{
        float value = 1;
        if (other.type == this.type)
        {
                // the type defines which dimensions can be compared
                if (type == BirdInformationType.FOOD)
                {
                        // the vector containing the comparable
                           information of this
                        float[] infoVectorA = new float[]
                        {
                                this.age,
                                this.foodSourceSize,
                                this.foodSourcePosition.x,
                                this.foodSourcePosition.y,
                                this.foodSourcePosition.z
                        };

                        // the vector containing the comparable
                           information of other
                        float[] infoVectorB = new float[]
                        {
                                other.age,
                                other.foodSourceSize,
                                other.foodSourcePosition.x,
                                other.foodSourcePosition.y,
                                other.foodSourcePosition.z
                        };

                        int foodProperties = infoVectorA.Length; //
                           the number of comparable food properties
                        float[] infoVectorDistance = new
                           float[foodProperties];
                        float sqrMagnitude = 0;
                        for (int i = 0; i < foodProperties; i++)
                        {
                                infoVectorDistance[i] =
                                   infoVectorA[i] - infoVectorB[i];
                                sqrMagnitude += infoVectorDistance[i]
                                   * infoVectorDistance[i];
                        }
                        value = Mathf.Sqrt(sqrMagnitude);

                        // to make indifferent to environment
                           circumstances, calculate value in range
                           (0,1)
                        value /= maxFoodDifferenceVectorMagnitude;
                }
        }
        return value;
}
```

4.7 Data Gathering for Measurements

The environment manages data gathering for measurements. An object of type Measurement saves important information and writes it to a file after the simulation has ended. Relevant information is stored during each measurement update. This includes:

1. The number of hungry birds
2. The number of feeding birds
3. The number of communicating birds
4. The number of exploring birds
5. The number of discovered food sources
6. The number of spawned food sources
7. The number of removed food sources
8. The average amount of time a food source is undiscovered
9. The average number of hops a piece of information has passed
10. The average number of duplicates existing per piece of information
11. The average age of a piece of information
12. The average certainty of a piece of information

The numbers used in items one to four are gained by keeping track of birds changing their states. This means that the sum of all three state counts can be different to the number of birds. These numbers keep track of how many birds have changed their states to the particular state in question since the last measurement update. The frequency of state changes can be derived from these numbers.

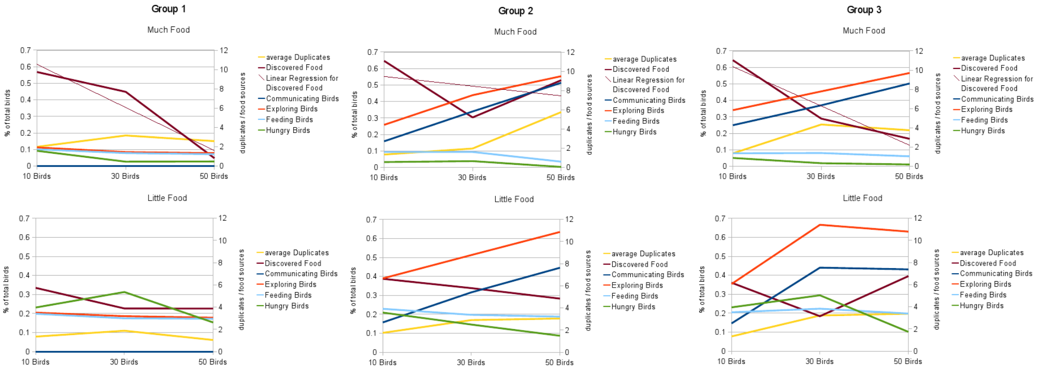
The next three items are straightforward. The environment is simply keeping track of the number of food sources.

Item eight keeps track of all undiscovered food sources' life timer and takes the average of this value.

Item nine gets the average of all birds' information's hops.

For item ten, the food source's id is stored along with the other attributes in the gathered information piece. It is only used for measurement purposes and does not influence a bird's functioning. The duplicate count is needed to know how fully a swarm is informed about the environment. The ideal duplicate count would be the same as the number of swarm-members, because then all members would know exactly the same. This situation has not occurred during any simulation run.

Items eleven and twelve store the average age and certainty of a piece of information.



(a) Group 1 Data (b) Group 2 Data (c) Group 3 Data

Figure 7: An overview of the collected data.

5 Evaluation

The results from testing and measuring the simulated artificial swarm are presented in this chapter.

5.1 Measurement

The way in which the measured data was acquired has been described in chapter 4.7. The different settings used to gain comparable data are discussed in the following section. The tools used for analyzing the results are presented after that.

The two variables which can be changed to create different circumstances are the amount of food and the number of birds.

The amount of food can be given in relation to the birds' sum of capacity for food (in relation to the size of their collective stomach, so to speak). If this value is one, there is exactly as much food in the environment as the birds can eat. There were two different settings used for the food variable. The first was set to half of what the birds can eat and the other to the double amount of food compared to the birds' food eating capacity.

For the simulations, swarms with ten, thirty, and fifty birds were evaluated.

In addition to the given variables, there are the three types of birds to be tested.

The length of one simulation period is fourteen minutes. The measurement update interval is five seconds, creating a data set with a length of 166 samples.

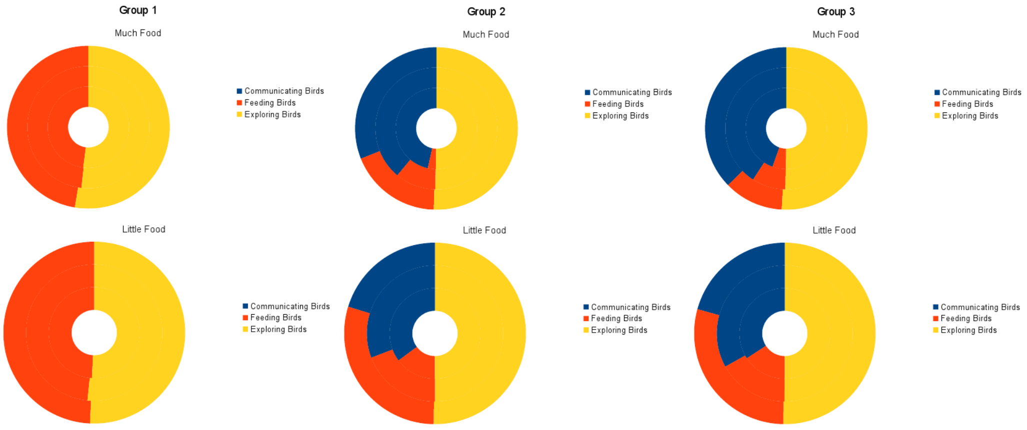
5.2 Results

The results from the simulation are presented in this chapter. The different relationships between defined attributes of the swarms which have been observed are discussed and analyzed. An overview of the gathered data can be seen in figure 7.

The data series called *Average Duplicates* refers to how many duplicates per information are in circulation at the time of measuring. If there are two pieces of information owned by birds which have the same food source as subject, there is one duplicate information in circulation. This data series uses the second y-axis on the right-hand side of the diagrams.

The data series called *Discovered Food* relates to the number of food sources which have been discovered since the last measurement update has taken place. It also uses the second y-axis.

All following data series use the first y-axis on the left-hand side of the diagrams.



(a) Group 1 Data (b) Group 2 Data (c) Group 3 Data

Figure 8: The average state changes per amount of birds. The inner ring represents ten birds, the middle ring 30 birds and the outer ring 50 birds.

The data series called *Communicating Birds*, *Exploring Birds* and *Feeding Birds* are percentages of the total amount of birds. These data series show how many birds changed to a certain state since the last measurement update. This means that if a bird has changed to the exploration state twice, it is counted twice and if a bird has not changed its state at all, it is not counted and does not show up in any of the series. These series show the amount of activity in a swarm.

The data series called *Hungry Birds* shows the percentage of birds which have been hungry in the last five seconds (the measurement update interval).

An overview of the activity is shown in figure 8. The inner ring represents activity data for ten birds, the middle ring represents the data for thirty birds and the outer ring represents the data for fifty birds.

5.2.1 Relationship between Information Gathering and Amount of Hungry Birds

The activity of group 1 is not enhanced by more birds in the swarm. The amount of birds changing to the exploration and feed states is nearly the same for this group.

For all three groups, the number of birds changing to the exploration state is bigger than the amount of birds changing to the communication state. Both data series exhibit a similar increase or decrease in birds changing to the according states. The Hungry Bird data series shows a trend towards decreasing amounts of hungry birds with growing numbers of birds belonging to the swarm. Another peculiar trend is the one of the two information gathering data series towards increasing with the number of birds in a swarm.

The activity of a swarm rises with the number of birds in it. This might be due to the increased density of birds in the environment. The environment's bounds are constant for all simulations. When the density is bigger, the probability of a bird to meet another bird increases. This can be seen in the enhanced activity visualized by the data series called Communicating Birds. Birds switch back to the exploration state after having communicated. Therefore the Exploring Birds data series is linked to the Communicating Birds data series.

Another effect of high density swarms is that they can execute the surveillance of an area more efficiently. This can be seen in the Hungry Birds data series, which decreases with the total number of birds in a swarm.

In a high density swarm, spreading of information can happen with a higher probability. The more birds there are in the swarm, the more duplicates of information pieces can be observed.

The connection between information gathering and the amount of hungry birds is established by the density of a swarm. The higher the density, the easier it is for birds to spread information and the easier it is to fulfill their primary goal of not being hungry.

5.2.2 Differences between Much and Little Food Simulations

The most obvious difference between these two types of simulation is that with less food there are more hungry birds. This is due to the lower chance of finding food. When food has been found and the information about this food source is given to another bird, there is a high chance of this food source already being fully depleted. Birds are more likely to follow wrong information.

Another difference is that when there is less food in the environment, birds do not have as much information to pass on and because they are hungrier than when there is much food, they do not change into the communication state as often. This causes the bigger gap between communication and exploration activity when less food is involved.

5.2.3 Relationship between Number of Birds and Information Duplicates

In groups 2 and 3, the duplicate count goes up with the number of birds. This can be explained with the bigger amount of birds communicating (see chapter 5.2.1). The more birds communicate, the more often they come across information which they did not already know. This new piece of information is copied and thus creates a duplicate. The information of the food source itself is not counted as a duplicate.

In group 1, the duplicate count goes down with the number of birds. The reason for this could be that the activity of group 1 does not include communication. The only state with a high cohesion force is the communication state. The birds from group 1 are either in the exploration or in the feed state. This keeps the birds fairly evenly distributed over the environment, which has the effect that birds do not often come across the same food source. Thus the small number of duplicates.

5.2.4 Comparison of Group 1 to the Other Two Groups

Group 1 exhibits significantly less state changes than the other two groups. This is because of the missing ability to talk. The only occasion when birds of this group change their state is when they are hungry and know where to find food.

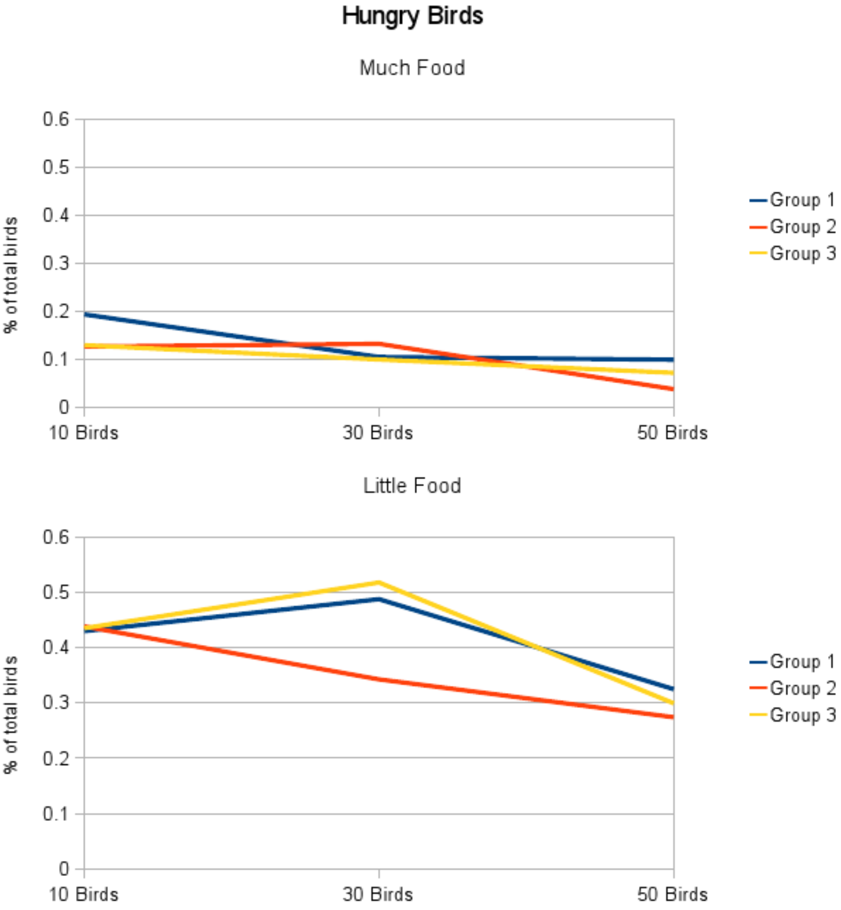
The numbers of hungry birds are visualized in figure 9. This figure shows the outcome of all actions taken by a swarm to keep fed. The main goal of the swarms has been defined as having as little birds hungry as possible. This diagram shows the success of the three test groups in reaching that goal.

Group 1 performs well with 30 birds. This equates a density of 0.00024 birds per cubic units. When this group is tested with an environment containing much food, it performs worse than the other two groups.

5.2.5 Comparison of Group 2 to Group 3

When there is little food in the environment, group 2 performs better the more birds there are. Group 3 has a similar performance to group 1 when there is little food. When there is much food in the environment, group 3 performs significantly better than group 1 and 2 for up to approximately forty birds. As of fourty birds, group 2 has less hungry members than the other two groups and is therefore better equipped for this kind of challenge.

Figure 9: The amount of hungry birds for each test group. The upper diagram shows an environment with much food and the lower diagram an environment with little food.



6 Conclusion

Three different types of swarms have been implemented and tested under different conditions. The types of swarms can be discerned by their communication abilities.

Test group 1 was not able to communicate with other birds actively. The observed effects of this disadvantage were a less often changing of states, generally more hungry birds, and therefore a smaller skill in fulfilling the swarm goal of keeping as many birds fed as possible. An advantage of this group is its ability to keep information duplicates low. If there is no need for every bird to know everything, this ability might prove useful.

Test group 3 has been shown to perform well in an environment with much food and up to approximately forty swarm members. This group does not maintain as many duplicate information pieces as group 2 due to its certainty limitations. There are not as many state changes to the communication state as in group 2, thus saving time for exploring the environment. Related to fulfilling its primary goal, group 3 performs worse than test groups 1 and 2 in a sparsely resourceful environment. Group 3 exhibits behavior which leans towards the exploration side in an environment with little food.

Group 2 shows the best performance regarding little food environments. Group 2 shows a relatively high rate of state changes to the feed state in this kind of setting. This means that wrong information is falsified and removed earlier than in groups 1 and 3. Group 2 birds communicate more often with other birds than agents of group 3. In group 2, there are many duplicates of information in circulation in an environment with much food.

The main difference between group 2 and group 3 is the way in which information is falsified. In group 3, possibly wrong information is not taken into account and discarded from the first moment on. In group 2, the falsification happens via exploration. When a bird wants to eat food which is not there anymore it discards the information. This is done by both groups, but test group 3 does not deem as much information important as the other group.

It can be concluded that prematurely discarding information leads to a smaller information base and therefore to more potentially wrong decisions. The time needed for talking to others and the time needed for confirming whether information is accurate or not do not outweigh the advantages of a bigger information base.

List of Figures

Listings

List of Tables

References

Bonabeau, Eric, Marco Dorigo, and Guy Theraulaz. 1999. *Swarm intelligence from natural to artificial systems.* 1 –182. Oxford University Press.

Dimeas, Aris, and Nikolaos Hatziargyriou. 2007. Agent based control of virtual power plants. *International Conference on Intelligent System Applications to Power Systems, ISAP 2007* 14:536 –541.

Ghanem, Bernard, and Narendra Ahuja. 2011. Modeling dynamic swarms.

Gloor, Peter. 2006. *Swarm creativity.* 22. Oxford University Press.

Kennedy, James, and Russel C Eberhart. 2001. *Swarm intelligence.* Academic Press.

Li, Ming, Kejie Lu, Hua Zhu, Min Chen, Shiwen Mao, and B Prabhakaran. 2006. Robot swarm communication networks: architectures, protocols and applications. *International Conference on Communications and Networking* 3:162 –166.

Reynolds, Craig. 1987. Flocks, herds, and schools: a distributed behavioral model. *SIGGRAPH Computer Graphics* 21:25 –34.

Schill, Felix, and Uwe Zimmer. 2006. Effective communication in schools of submersibles. *IEEE Oceans* 1:1 –5.

Schill, Felix, and Uwe R. Zimmer. 2007. Distributed dynamical omnicast routing. *Complex Systems* 16:299 –316.

Vytelingum, Perukrishnen, Thomas D. Voice, Sarvapali D. Ramchurn, Alex Rogers, and Nicholas R. Jennings. 2010. Agent-based micro-storage management for the smart grid. *International Conference on Autonomous Agents and Multiagent Systems* 9:39 –46.

7 Appendix A - Source Code

The source code can be found on GitHub at `http://github.com/junglejinn/bakk2`.

8 Appendix B - Measured Data

On the next few pages are the averages of the measured data which were used in chapter 5.

Table 3: Averages of the measured data, Little food environment

10 Birds	Little Food			
		Certain Info Bird	Gossip Bird	Not Talking Bird
	Hungry Birds	4.3473053892	4.377245509	4.2934131737
	Feeding Birds	2.0419161677	2.2874251497	1.9820359281
	Communicating Birds	1.4550898204	1.5688622754	0
	Exploring Birds	3.5329341317	3.8922155689	2.0419161677
	discovered Food	6.1796407186	6.6347305389	5.7365269461
	spawned Food	2.2814371257	2.1796407186	2.4491017964
	removed Food	2.5269461078	2.5568862275	2.5329341317
	average Discovery Duration	152.8839444192	182.5589229042	147.3080776407
	average Hops	0.0192973428	0.0947791924	0
	average Duplicates	1.3184083892	1.7467041916	1.3395708503
	average Information Age	36.4656	27.1451498323	26.3460888323
	average Information Certainty	0.9995175401	0.996686621	1
30 Birds	Little Food			
		Certain Info Bird	Gossip Bird	Not Talking Bird
	Hungry Birds	15.5209580838	10.2694610778	14.6167664671
	Feeding Birds	6.6946107784	5.9041916168	5.2335329341
	Communicating Birds	13.1976047904	9.4610778443	0
	Exploring Birds	19.9760479042	15.3652694611	5.5748502994
	discovered Food	9.3293413174	12.4251497006	9.6167664671
	spawned Food	3.8263473054	4.4011976048	4.0179640719
	removed Food	5.4610778443	8.3353293413	5.6766467066
	average Discovery Duration	221.0292068443	291.5991038931	213.0519703299
	average Hops	0.132418939	0.1257033676	0
	average Duplicates	3.2092388554	2.9000533952	1.8704553892
	average Information Age	23.2729811078	32.506035491	20.5010822934
	average Information Certainty	0.9880535042	0.9949158449	1
50 Birds	Little Food			
		Certain Info Bird	Gossip Bird	Not Talking Bird
	Hungry Birds	14.9401197605	13.6946107784	16.2275449102
	Feeding Birds	9.8862275449	9.3652694611	8.5988023952
	Communicating Birds	21.5361445783	22.2634730539	0
	Exploring Birds	31.502994012	31.7005988024	8.8862275449
	discovered Food	16.1077844311	17.2694610778	13.4850299401
	spawned Food	6.2634730539	6.6826347305	6.1437125749
	removed Food	14.4491017964	15.3892215569	13.245508982
	average Discovery Duration	320.8988920958	329.4713479042	316.2711639401
	average Hops	0.2230305237	0.0640268634	0
	average Duplicates	3.377340494	3.0409475241	1.0375244731
	average Information Age	39.1714155856	44.4178158677	43.7084863353
	average Information Certainty	0.9852677892	0.9913860168	1

Table 4: Averages of the measured data, Much food environment

10 Birds	Much Food			
		Certain Info Bird	Gossip Bird	Not Talking Bird
	Hungry Birds	1.2934131737	1.2634730539	1.9341317365
	Feeding Birds	0.7904191617	0.9401197605	1.0119760479
	Communicating Birds	2.4910179641	1.5868263473	0
	Exploring Birds	3.4011976048	2.5808383234	1.125748503
	discovered Food	11.0359281437	11.0838323353	9.7365269461
	spawned Food	2.4850299401	2.4790419162	2.5628742515
	removed Food	6.245508982	6.1916167665	5.6107784431
	average Discovery Duration	246.2162216054	266.2037225689	224.0987041677
	average Hops	0.0661606475	0.0461381924	0
	average Duplicates	1.3448768024	1.3460411317	1.9889196168
	average Information Age	107.3025727066	91.6925818982	69.4814653533
	average Information Certainty	0.9978223168	0.9988326299	1
30 Birds	Much Food			
		Certain Info Bird	Gossip Bird	Not Talking Bird
	Hungry Birds	2.9880239521	3.9640718563	3.1497005988
	Feeding Birds	2.4191616766	2.8203592814	2.3532934132
	Communicating Birds	11.0898203593	10.1916167665	0
	Exploring Birds	13.6407185629	13.1556886228	2.5329341317
	discovered Food	15.9880239521	16.2874251497	17.4251497006
	spawned Food	3.6467065868	4.0239520958	3.7904191617
	removed Food	17.7784431138	16.2634730539	19.0479041916
	average Discovery Duration	340.0202806167	330.4413960713	343.7979169263
	average Hops	0.056841716	0.3781110115	0
	average Duplicates	4.3425464192	1.9677753054	3.1700462515
	average Information Age	129.4027241796	102.6335989707	124.7409094132
	average Information Certainty	0.9988220299	0.9861341138	1
50 Birds	Much Food			
		Certain Info Bird	Gossip Bird	Not Talking Bird
	Hungry Birds	3.5988023952	1.9041916168	4.9580838323
	Feeding Birds	3.0359281437	1.7425149701	3.5868263473
	Communicating Birds	25.125748503	25.6766467066	0
	Exploring Birds	28.3053892216	27.6646706587	3.8862275449
	discovered Food	18.8443113772	25.3592814371	18.2335329341
	spawned Food	4.0658682635	5.2694610778	4.1437125749
	removed Food	32.4610778443	42.502994012	29.9461077844
	average Discovery Duration	369.4699478802	363.8517209042	366.6203541677
	average Hops	0.2382044685	0.0256694254	0
	average Duplicates	3.7469057006	5.7552057126	2.5795545988
	average Information Age	142.0518726826	198.8493153533	114.6511073653
	average Information Certainty	0.9880881395	0.9987600299	1