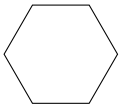
Master Thesis, Institute of Computer Science, Freie Universität Berlin

Biorobotics Lab, Intelligent Systems and Robotics

Temporal Analysis of Honeybee Interaction Networks

based on Spatial Proximity



Alexa Schlegel

Matriculation number: 4292909

alexa.schlegel@fu-berlin.de

Supervisor: Prof. Dr. Tim Landgraf, Freie Universität Berlin

Second Supervisor: Dr. Philipp Hövel, Technische Universität Berlin

Berlin, April 5, 2017

Eidesstattliche Erklärung

Ich versichere hiermit an Eides Statt, dass diese Arbeit von niemand anderem als meiner Person verfasst worden ist. Alle verwendeten Hilfsmittel wie Berichte, Bücher, Internetseiten oder ähnliches sind im Literaturverzeichnis angegeben, Zitate aus fremden Arbeiten sind als solche kenntlich gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungskommission vorgelegt und auch nicht veröffentlicht.

Berlin, den April 5, 2017

Alexa Schlegel

Abstract

TODO

Zusammenfassung

TODO

Contents

Chapter 1

Introduction

A social insect society is formed by thousands of individuals, which continuously move and interact with each other inside a dark nest. Honey bees are organized in colonies, which form a complex and dynamical system. Observing individual honey bees and their interactions with each other is, therefore, vital for understanding collective behavior and the organization of tasks within the colony.

Within the BeesBook project of the Biorobotics Lab of Freie Universität Berlin Wario et al. [38] developed technologies to automatically track all individuals of a honey bee (Apis mellifera) colony, that are inside the honeycomb. Shortly after hatching, each bee is marked on their thorax by using circular 12-bit tags (figure 1.1) and then added to the observation colony. Four cameras observe the comb over a period of nine weeks, by capturing approximately three frames per second. An image analysis pipeline evaluates each frame automatically. The resulting data set contains, for each frame, the exact position of each detected bee on the honeycomb, and its age.

In this thesis, worker-worker interaction networks, based on spatial proximity, are derived from the described data set. Each node in the network is a bee, and a link between two nodes results if two bees are located close to each other over a specified period. The networks are time-aggregated, which means one network represents the data of multiple frames. After extracting the temporal networks, social network analysis methods are applied to determine the characteristics of the resulting networks and its communities.

Figure 1.1: Tagged bees inside the observation hive.

1.1 Motivation

Colonies of social insects consist of a vast number of individuals. The technique of manual insect tagging and tracking is widely applied in the behavioral sciences: First animals are marked using colored paint or numbered tags to distinguish individuals. Then, they are observed using a video recorder or by taking photos. The interaction data is obtained by repeatedly watching the video files and manually extracting events. Consequently, labeling only a subset of the colonies individuals, a short observation period, a low number of frames, or limiting the observation to only a small area of the hive is very common. Accordingly, most studies in the field of animal social network analysis, related to insects, analyzed only a reduced subset of a colonies' life. The majority of social insect interaction networks studies, due to previously technical limitations, aggregate temporal tracking data into a single static network [19, Chapter 15].

Recently, automated tracking of insects has become technically feasible [38, 9, 12]. Using automated high resolution tracking data, which includes all individuals of the complete comb over an extended period allows for more advanced analysis focusing on temporal dynamics. Therefore, automatic tracking allows shifting more towards the temporal and dynamic investigation.

1.2 Research Goal and Method

The aim of this thesis is to investigate whether the provided data set of tracked honey bees is useful for creating worker-worker interaction networks using spatial proximity as an indicator for interactions between bees. Thus, I need to implement a pipeline to extract networks out of the given data set. Furthermore, I want to find out if the resulting networks are suitable for social network analysis.

I want to achieve my research goals by answering the following questions:

1. *Is it possible to infer temporal networks with the provided honey bee tracking data?*
   What challenges and limitations does the data set imply? What pipeline parameters are necessary?
2. *What kind of worker-worker interaction networks emerge and how are they structured?*
   What is their topology? What properties are characteristic and how do they differ from randomly generated networks?
3. *Does the network display a meaningful community structure?*
   How are the identified communities characterized? Do they reflect already known colony behavior concerning age and spatial distribution?
4. *How do these communities develop over time?*
   Are they stable regarding their properties? How do members move between communities?

This work is meant to be the foundation to answer further more specific biological research questions using a network science approach to study the complex system of honey bee colonies and their collective behavior.

The methodology of this work consists of two parts, described in detail in Chapter 3. The first part deals with the approach to infer and define spatial proximity networks using the given tracking data of honey bees. It serves as a prerequisite for analyzing the resulting networks concerning its network properties, communities and its development in the second part.

1.3 Outline

This thesis is organized as follows. Chapter 2.1 gives a short introduction into social network analysis (SNA) and defines network measures, terms, and algorithms used throughout this work. In chapter 2.2, a brief summary of the current state of research concerning social insect networks, temporal networks and community detection in animal social networks is given. Chapter 3 describes my research approach in general and how the pipeline infers networks out of the given dataset, what steps are needed and what parameters it uses. Also, I explain and justify what decisions I took during the network analyses and community detection process. Chapter 4 reports the results of the network analysis and the characteristics of the extracted communities. Finally, in chapter 5 I explore the results, discuss limitations and conclude with directions for future work.

Chapter 2

Network Analysis of Insect Colonies

TODO some intro

2.1 Social Network Analysis

The following chapter gives a short introduction into social network analysis (SNA). It introduces social insect interaction networks, as a special type of a biological[1] network. Also, it defines terms and concepts used throughout this work and explains used network metrics and algorithms.

A *social network* is a representation of a social structure comprising actors such as individuals, affiliations, as well as their social interactions. The network model conceptualizes social, economic, or political structures as lasting patterns of interactions between actors [39]. In mathematical terms, networks are graphs, and thus consist of *nodes* (vertex, representing individuals), and *links* (edges, relationships or interactions). Social network analysis provides a set of methods, measures and theories, borrowed from network and graph theory, to investigate social structures and its dynamics.

This work is focusing on the special case of social insect networks, where individuals are nodes and edges are defined as interaction events between individuals are called *interaction networks*, sometimes called association networks. According to Charbonneau et al. [8] those interactions used as an edge can be of four different types when looking at social insect networks: spatial proximity, physical contact (usually with antennae, "antennation"), a food exchange event (trophallaxis), or specific communication signals.

Edges can be directed (e.g. trophallaxis) or undirected, weighted or unweighted. The edge weights represent the strength of the relationship; commonly the number or duration of interactions is used [11].

[1] Maybe more preciese: within species interaction network.

2.1.1 Temporal Networks

When modeling temporal or so-called dynamic networks two main approaches exists (1) time-aggregated (discrete), where the data is aggregated either in a disjoint, overlapping or cumulative snapshot, and (2) the time-ordered (continuous) approach, with interactions having a start and end timestamp [21, 29, 6].

The time-aggregated approach aggregates the data for each snapshot and therefore reduces the available information per edge. In contrast, the time-ordered approach keeps the information for each edge, when the interaction occurred and how long it lasted. It provides a detailed insight when timing and order of interactions are important. And therefore it can be used to model the topological flow information through a network.

Choosing suitable time intervals for aggregating is challenging [29], but a lot of methods for analyzing those networks already exists, whereas for time-ordered networks, only limit toolset is available. In time-aggregated networks, the modeling nodes and edge weights can be challenging when taking into account that interactions, which took place earlier or later in time are weighted accordingly.

2.1.2 Network Measures and Metrics

The following definitions are mainly taken from Barabási [2] and Newman [25]. the gray box summarizes the basic variables and terms of this work. [TODO: Box referencing as table and align bottom.]

Network size N is the total number of nodes, respectivly animals in a network.

Number of links L is the total number of links, social interactions, in a network.

Edge weight w_i of an edge l_i is an indicator of how important that edge is.

Component is a subnet of nodes in a network, so that there is a path between any two nodes that belong to the component.

Degree k_i of a node n_i represents the number of edges a node has; so the number of other animals this animal interacts with.

Average Degree $\langle k \rangle$, the number of animals one animal interacts with on average.

Path length d the shortest number of links between two nodes.

Average path length $\langle d \rangle$ is the average of all shortest path between all pairs of nodes.

Diameter $d_{\texttt{max}}$ is the longest of all path length. The distance between the two furthest nodes, the longest possible path length in the network.

Density D is the number of realized links divided by the number of theoretically possible links is defined as

$$D = \frac{2L}{N(N-1)}$$

Is it independent from the edge weights.

Strength s_i of a node is also called the weighted degree. It measures the total weight of edges connected to a node n_i and is definded as

$$s_i = \sum_{j=1}^{n} w_{ij}$$

according to Barrat et al. [4]. The average strength denoted as $\langle s \rangle$.

Global clustering coefficient C_Δ also called transitivity. According to Wasserman and Faust [39] it is defined as

$$C_\Delta = \frac{3 \times \texttt{number of triangles}}{\texttt{number of connected triples}}$$

.

Local clustering coefficient c_i of a node n_i quantifies how close its neighbours are to being a clique (complete graph).

Centrality When looking at the networks local structure (node level), it is possible to identify nodes, which are important or central to the network, regarding different aspects. This concept is called *centrality* and measures the influence of a node in a network. [25] In the course of analysing networks and their local node level structures, you will find and encounter the most important (central) nodes and vertices by indicators of centrality. These indicators give values to the nodes and therefore they can be listed in a way of importance.

The the weighted versions of betweenness and closeness using Dijktra and the inverse of the edge weights.

Degree Centrality Degree centrality C_D^i of a node n_i is the normalized degree k_i in relation to the whole network, it is calculated as follows:

$$C_D^i = \frac{k_i}{N-1}$$

Eigenvector Centrality The eigenvector centrality x_i of a node n_i is the sum of its connections to other nodes, weighted by their centrality.

$$x_i = \frac{1}{\lambda} \sum_j A_{ij} x_j$$

It is like a recursive version of degree centrality. So a nodes importance (centrality) is inceased by having neighbours that are themselves important. Eigenvector centrality gives each vertex a score proportional to the sum of the scores of its neghbours. [25]

Closeness Centrality Is is the average length of the shortes path between node n_i to all other nodes in the network. The more central a node is the closer it is to all other nodes. Mean distance from a node to other nodes. [25]

$$C_C^i = \frac{N}{\sum_j d_{ij}}$$

Betweenness Centrality It measures the extend to which a node lies on paths between other nodes. Nodes that occur on many shortest paths between other nodes have higher betweenness that those that do not.

2.1.3 Community Detection

To understand the large-scale structure of networks, one can look at the network's community structure. Communities are naturally occurring groups within a network, usually also called clusters, cohesive groups or modules and have no widely accepted, unique definition [28]. For my work, I adapt the definition according

to Barabási [2]: "In network science, we call a community a group of nodes that have a higher likelihood of connecting to each other than to nodes from other communities." [2, p. TODO]. In contrast to a simple graph partition, the number and size of communities is not predetermined or set in advance.

Communities in animal social networks refer to groups of individuals that are associated more with each other than they are with the rest of the population. These communities reflect an intermediate level of social organization, which is located between the individual and population level [10].

There are a lot of different approaches and algorithms who adress the detection of communities. Fortunato [14] gives an extended overview of the various types of community detection algorithms. Explaining any of those would be beyond the scope of this work. For example, traditional methods include algorithms based on graph partitioning, hierarchical clustering, and spectral clustering. There are also divisive and agglomerative algorithms. The algorithms used in this work are described in the following sections and include the leading eigenvector [26] and walktrap [31] algorithm.

Modularity

Modularity is a quantity, that measures the quality of a partitioning. It can be used to compare a community partition to another and decide for the better one. Modularity optimization is also used for community detection algorithms.

A high modularity of a network indicates more connection between nodes within a community and fewer connections between nodes of different communities. The basic idea is: If the fraction of links inside the community is higher, than expected in the same community of a related random graph having the same degree distribution, then it is a community in the sense of modularity. This difference is summed up and normalized. If all nodes fall into one community the modularity is 0. Fewer links inside the community than expected result in a negative value, otherwise positive.

Leading Eigenvector and Walktrap

The *leading eigenvecor* algorithm was proposed by Newman [26]. It uses the eigenvectors of matrices for finding community structures in networks. It is a top-down hierarchical approach that optimizes modularity. The algorithm starts with all nodes inside one community, therefor a modularity of 0. In each step, the network is split into two parts, so that the modularity of the new separation increases. The splitting is done by first calculating the leading eigenvector of the modularity matrix and then splitting the graph in a way that modularity improvement is maximised based on the leading eigenvector. The algorithms stops if the modularity is not increasing anymore.

This *walktrap* algorithm by Pons and Latapy [31] is based on random walks. The authors consider random walks as a tool to calculate similarity between nodes of

a network. It uses a bottom-up hierarchical approch, that means the algorithms start with each node it is own community. The basic idea of walktrap is, that short distance random walks (the step size is a parameter) tend to stay in the same community, because there are only a few links that lead outside a given community. The results of these random walks are used to merge separate communities. Again modularity can be used to cut the dendrogram in an optimal place.

2.2 Related Studies

Relevant for my work are studies using a network analysis approach focusing on interaction networks[2] to investigate the behavior of social insects, especially honey bees. I mainly reviewed studies mentioned in the survey papers of Pinter-Wollman et al. [29], Krause et al. [19, chapter 15] and Charbonneau et al. [8].

The most relevant studies were classified by:

- type of analysis: temporal or static analysis (using automated or manual tracking over a long or short term); and
- studied species: honey bees or other social insects.

Additionally, I inspected their shortcomings regarding time, space, and the number of tracked individuals, and thus, examined the following characteristics: total duration of study, observation period, sampling resolution, the number of colonies and marked individuals and space limitations. Table A.1 (Appendix A) summarizes the selected studies and their characteristics. I also recorded whether the studies included the aspect of age cohorts in their analysis and I listed the used software tools for network analysis.

Within the scope of my literature review, I found a lot of studies in the field of static network analysis of ants [15, 30, 32, 13, 40, 36], wasps [22] and bumblebees [27], but only a few related to honey bees [3, 23, 35, 24]. Also, I found several studies focusing on temporal aspects of ant colonies [20, 5, 18], but I didn't find any for honey bee colonies.

2.2.1 Static Network Analysis of Honey Bee Colonies

The most advanced work studying honey bees using a network science approach is by Baracchi and Cini [3]. Their studie revealed a highly compartmentalized structure inside the honey bee colony: Depending on the age, bees occupy separate areas of the comb and perform different tasks. Also, there is limited contact between different age groups.
The frequency of interactions between bees is used as weights for edges in an undirected worker-worker interaction network. The body length of a bee defines the radius of spatial proximity. Baracchi and Cini make use of the node level measures strength (weighted degree), closeness and eigenvector centrality to investigate the networks. Furthermore, they perform a cluster analysis using the dissimilarity measures 'average linkage between groups' and 'squared Euclidian distance among network values.' The main drawback is that they marked only 211 bees from three predefined age cohorts out of one colony with 4000 individuals and observed only one side of the observation hive for ten hours by capturing with a low resolution of one frame per minute. [TODO: explain drawback of clustering in a better way, ask somebody!]

[2] Studies using worker-task, worker-nestarea, nestarea-nestarea or other bipartite networks are excluded.

Scholl and Naug [35] investigate the mechanism behind the emergence of organizational immunity of honey bee colonies by using unweighted, undirected physical contact and trophallaxis networks. In their case, the observation is limited to one hour per day, with three days of observation spread over three weeks. In the field of network analysis they investigated the interactions between three predefined age groups.

Naug [23] inspects the network structure of weighted, directed trophallaxis networks using four age cohorts. He evaluates the changes in transmission dynamics produced by experimental manipulation. The data set is limited to one hour and only first- and second-order trophallaxis interactions are considered.[3]

2.2.2 Temporal Network Analysis of Insect Colonies

Regarding the used methods, the study of Mersch et al. [20] is very close to my work. They automatically tracked all individuals of six ant colonies over a period of 41 days using a resolution of two frames per second. For each observation day, the authors extracted time-aggregated weighted contact networks per colony, using antennation as the physical contact event. They applied the Infomap community detection algorithm to each daily network and thus revealed three distinct and robust groups. Each group represents a functional behavioral unit, with ants changing groups as they age. Except for community detection, they did not use any other network science methods to investigate the network properties.

Another work, using automatic tracking, is byJeanson [18]. It focuses on the investigation of the temporal stability of spatial proximity networks in four ant colonies over three weeks. Here, proximity is defined as $\frac{4}{3}$ of an ant's body length. Per week and colony they generated weighted time-aggregated networksy, using the total duration of interaction as the edge weights. They investigated the strength, betweenness and closeness centrality and found out that the networks are stable over time, without the queen contributing to the network structure. Also they state that individuals with long lasting interactions seem to have a reduced tendency to move, while mobile ants interact homogeneously with their nestmates. The size of the observed colonies ranges from 55 to 58 individuals.

The only study not only using time-aggregated but time-ordered (dynamic) networks is by Blonder and Dornhaus [5]. They marked all individuals of four ant colonies and filmed each colony for 30 minutes on two days with three weeks in between. The interaction events, physical contact of antenna and body, were manually extracted by watching the videos. In the resulting networks, the edges are time-stamped interactions between individuals. Their study shows how temporal and spatial dynamics of individual interactions provide upper bounds to rates of colony-level information flow and how this flow scales with individual mobility and group size. [TODO: reframe summary of study results! ask somebody] This very specialized study on dynamics in information flow also observed colonies with 6 to 90 individuals.

[3]The food transfer from the forager to a worker bee is called first level interaction, the food transfer from that worker bee to other bees is called second-order.

In all these three studies each of the observed ant colonies contained a maximum of 200 individuals. This number is relatively small compared to the size of honey bee colonies used in the static analysis approaches.

Chapter 3

Methodology

This chapter describes the workflow and implementation I applied to reach my research goals. In the first section, I describe the given data set and the approach to infer networks. This first step of network interference, was primarily driven by a combination of an exploratory data analysis and an iterative pipeline development processes. It serves as a prerequisite for the further thesis. The second section explains the methods I used to analyze the resulting networks regarding network properties, communities, and its development.

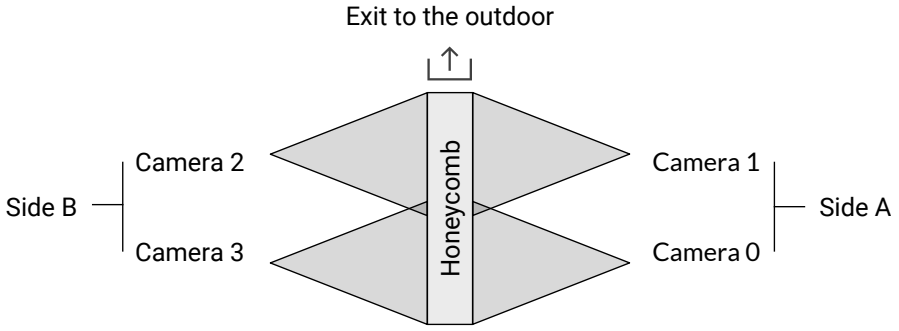
3.1 Inferring Spatial Proximity Networks

To yield the first set of functional and non-functional requirements concerning the pipeline, I conducted (1) a data analysis of the given tracking data, and (2) a literature review, already mentioned in chapter 2.2. The data analysis supported the forming of a general understanding of the given dataset, its structure, characteristics and estimation of its quality. The purpose of (2) was to get an overview of the common methods and approaches regarding network analysis in this field of research. The results of (1) and (2) are then used to decide for a type of network and its node and edge definitions. Furthermore, pipeline parameters are concluded, and I decided for the procedure of network extraction.

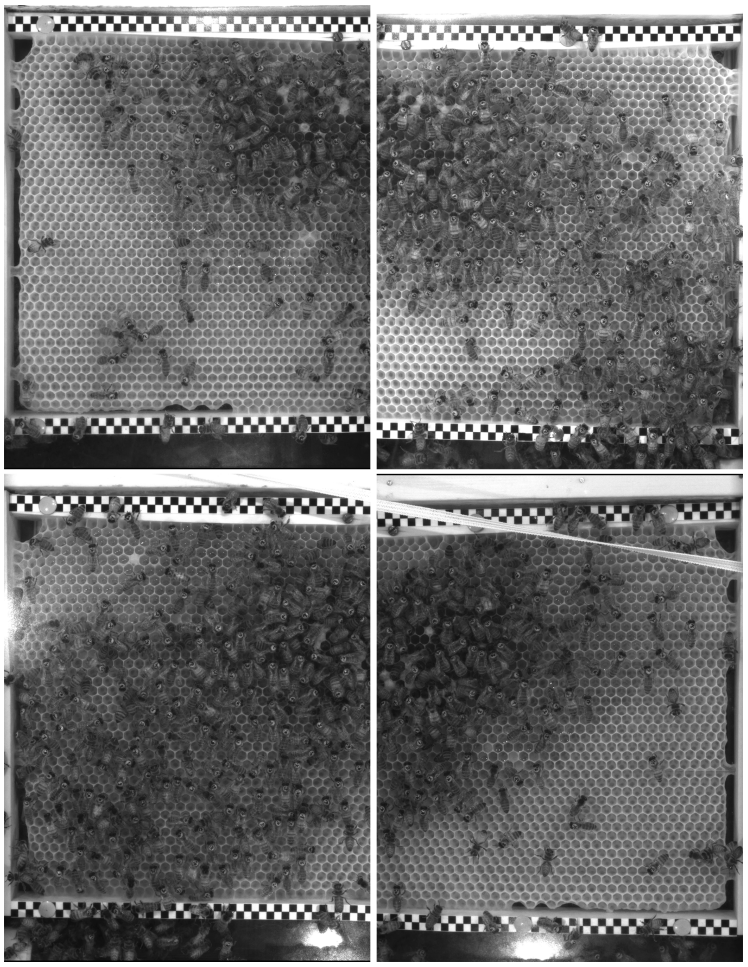
This pipeline was developed, tested and then refined in an iterative process. Accordingly, the results of the evaluation lead to new or changing functional requirements. The evaluation is conducted by reviewing the pipeline parameters' effects on network properties and checking the validity and quality of the networks by investigating the age of bees in the resulting network.

3.1.1 Describing the Dataset

The dataset derives from high resoluted video files, that capture tagged honey bees of one colony in an single-frame observation hive. The colony includes about 3200 bees over a period of nine weeks. The bees are uniquely tagged with circular 12-bit markers (figure 1.1, section 1). Two cameras per side filmed the complete honeycomb permanently. Figure 3.1a illustrates the camera setup. The *recording period* lasted



(a) Camera setup Each side of the honeycomb is filmed by two cameras. The two cameras per side overlap, bees inside this area are detected from both cameras.



(b) Images for each camera Top: side A, bottom: side B

Figure 3.1: Observation setup

nine weeks (63 days), from 19.07.2016 until 19.09.2016, with some interruptions due to maintenance work and technical failures. An overview about the complete reocrding period is given in figure A.5 in appendix A.

All four cameras, each with a resolution of 4000×3000 pixels, record 3.5 frames per second. An image analysis pipeline [38] detects all bees in each frame. The resulting detection data is stored in a binary file format. An python library[1] provides an frame-level access to those binary files. The size of the dataset is 470 GB, about 7.5 GB of binary data per day.

The 67 days long *tagging period* started on 28.06.2016 and lasted until 02.09.2016, resulting in 3.191 tagged bees. The young bees, which were raised in a separate incubator, were tagged and then added to the observation hive, about noon each day. Figure A.4 (Appendix A) shows the frequency of tagged bees per day. The hatching day for each bee is documented; therefore the age of each bee at a particular point in time can be calculated.

For further analysis, I chose three days: 20., 22., and 24. August. On the one hand, because the three days are evenly distributed (always two days between) and no data is missing. On the contrary, at his point in time, the hive also contained older bees which are likely to be foragers. Also, about 100 young bees were added to the colony.

[1] The library is called `bb-binary` and is created by the Biorobotics Lab. It can be found on GitHub: `https://github.com/BioroboticsLab/bb_binary`; Last accessed: 2106-02-16, 04:28PM

Frame container	A container for all frames, which belong to a specific video file of a certain camera.
Frame	This includes all detections of one camera image at a certain point in time.
Detection	A detection of a bee at a certain point in time.
Decoded ID	Identifier of a bee consisting of 12 probability values, representing 12 bits.
Confidence	Value between 0% and 100%.
ID	The decimal representation of an decoded ID, after applying a certain confidence value.
Bee time series	Binary sequence, indicating the absence and presence of a certain bee in a particular time interval.
Pair time series	Binary sequence, indicating the absence and presence of two bees in a particular time interval.

Data Scheme

The data is organized in so-called *frame containers*. Each frame container corresponds to one video file of a single camera and consists of about 1024 *frames*. So the frame container specifies the camera (*camId*), which recorded the video. Each frame holds a list of bees, which were detected by the image analysis pipeline and is attributed with a *timestamp*.

A bee *detection* has, among others, the following attributes:

xpos:	x coordinate of bee with respect to the image in pixel
ypos:	y coordinate of bee with respect to the image in pixel
decoded ID:	decoded 12-bit ID
cam ID:	ID of the camera $0, 1, 2, 3$
timestamp:	unix timestamp with milliseconds

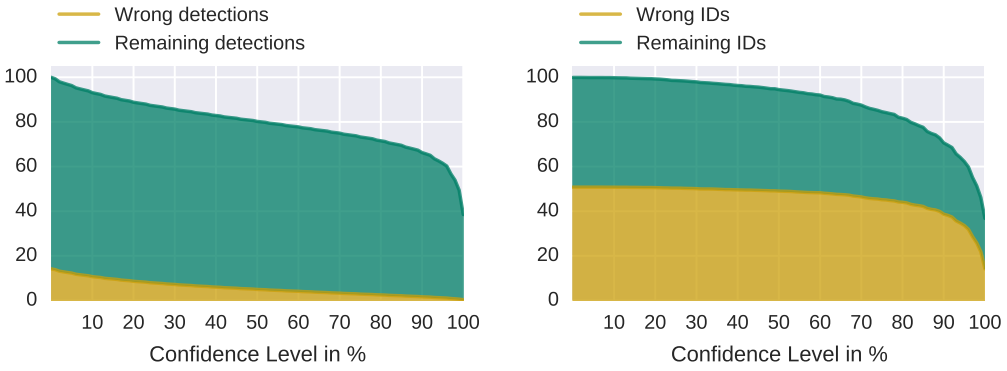
The data can be accessed by iterating on the frame level, using a start and end timestamp for specifying a time interval. The complete data scheme can be found on GitHub[2].

ID Probabilities, Confidence Level, and Quality

Twelve bits can encode the identity of 4096 bees. Each bit of the decoded ID is not a one or zero but represents a probability between 0 and 255, normalized to a value between 0 and 1. Therefore, a bit indicates the confidence of the image analysis pipeline for that specific bit. I define the confidence c for a bit b, analogously to Leon Sixt [37, p. 14], as $c(b) = 2 \cdot |b - 0.5|$. The confidence of a decoded ID is,

[2] `https://github.com/BioroboticsLab/bb_binary/blob/master/bb_binary/bb_binary_schema.capnp`; Last accessed: 2106-02-16, 04:46PM

[3] Data set: 26.07.2016, 4 p. m., 10 minutes, all cameras



(a) Detections **(b) IDs**

Figure 3.2: Quality of detections and IDs *Green* represents the number of remaining detections and remaining IDs (from 4096 possible IDs). *Yellow* indicates the fraction of wrong IDs and wrong detections in relation the remaining number of IDs and detections.[3]

accordingly, the minimum of all twelve bits' confidences. Consequently, a high level of confidence reduces the amount of data, which remains for further processing.

I use the age information of the bees to check the quality of the remaining data. The age is specified in days. An age of 0 days indicates that a bee was born on that day. It is possible that the resulting age is below 0 days. One the one hand, this happens when the pipeline detected a bee, that was not born yet. On the contrary, this can happen, if it discovered a bee tag, that was never used during the study, then the age is set to -100 days.

I examined (1) the number of detections and (2) the number of unique IDs, depending on the chosen confidence.

For (1) I calculated the age of each bee detection. A detection with a negative age is counted as a *wrong detection*. I assumed that a similar number of wrong detections also occurred among detections with a positive age, but remained unseen; therefore I doubled the error[4]. For (2), analogously a unique ID with a negative age is counted as a *wrong ID*. The total amount of wrong IDs is doubled.

As expected, with increasing confidence, the number of remaining detections and the amount of remaining unique IDs decreased (figure 3.2). Also even though the number of wrong detections decreases steadily with an increasing confidence level, the number of wrong IDs only starts to decline with a very high level of confidence. With a confidence level of 100%, 30.2% of the remaining unique IDs are invalid, corresponding to only 2.5% of invalid detections.

Therefore, to obtain a more reliable dataset, invalid detections need to be filtered out, independently of the confidence value. The amount of data that remains for further processing is still highly dependent on the chosen level of confidence.

[4]I chose the 26.07.2016 for testing this because half of the bee tags (2014 out of 4096) were already assigned.

Time Series of Bees and Bee Pairs

The dataset, is transformed to binary *bee time series*, depicted in figure 3.3 (left and middle). A time series of a bee is a sequence of zeros and ones indicating the absence and presence of a bee over a specified time interval. I examined the effect the level of confidence has on the bee time series. As expected, with an increasing confidence level the average gap length decreases and the overall number of gaps increases (figure 3.4).

The number of gaps, those bee time series has, is important because in a later step I want to extract pairs of close bees, who are present at the very same time. I call those *pair time series*, as shown in figure 3.3 (right). So a lot of gaps in bee time series could lead to a lot of gaps in the pair time series.

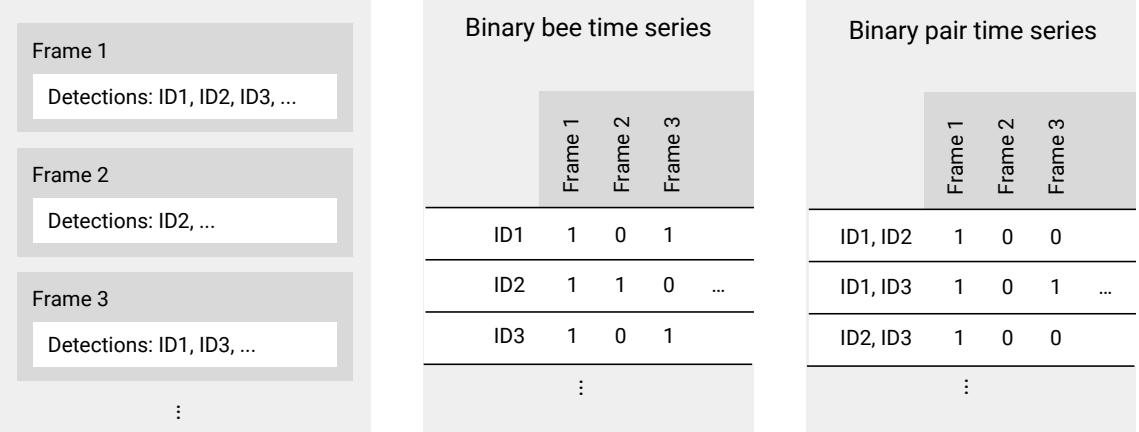
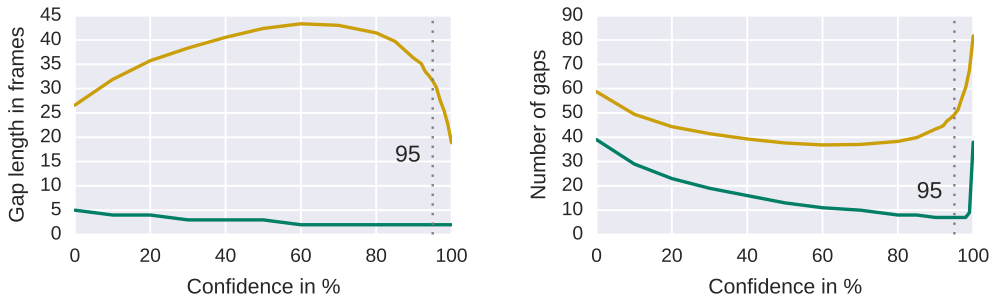


Figure 3.3: Structure of dataset *Left*: original dataset - containing a sequence of frames with bee detections; *Middle:* binary bee time series - zero and one indicate absence and presence of a bee; *Right:* binary pairs time series - zero and one indicate the absence and presence of two bees in the same frame.



(a) Length of gaps **(b) Number of gaps**

Figure 3.4: Influence of confidence level on gaps [TODO: add legend][TODO:gaps D] With an increasing level of confidence the average gap length decreases and the number of gaps per bee series increases. *Orange* indicates the median, *green the mean.*[5]

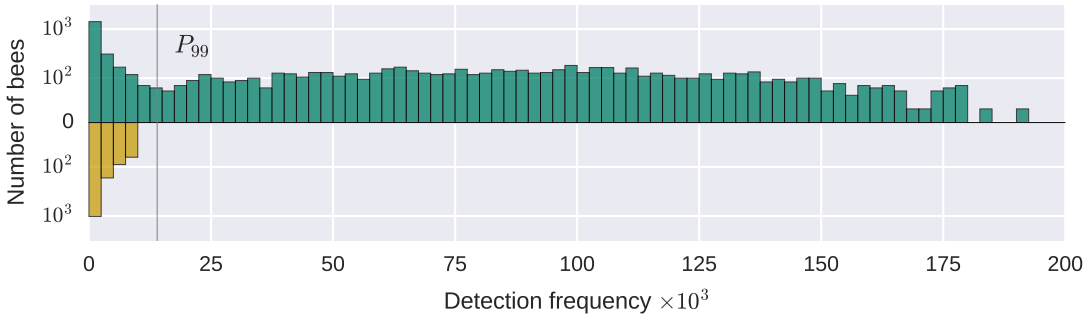


Figure 3.5: Detection frequency of IDs [TODO: add legend] *Orange* coressponds to bees with a negative age and *green* displays bees with a positive age.[6]

Detection Frequency Filter

A good indicator, whether a bee is alive and present on a particular day, is its detection frequency. The hypothesis is: Bees who have a low detection rate are not physically present in the hive, respectively do not exists at that day. To check this hypothesis, I investigate whether there is a correlation between the age of bees and their detection frequency.

Bess with a negative age are on average detected less frequently than bees with a positive age. In advance, I excluded bees from the statistics, which had a negative age, but a detection frequency over 10000 frames. Their detection frequency is similar to that from bees with a positive age. I looked at the corresponding photos taken by the camera and confirmed that those are living bees and no artifacts.[7] Also I excluded bees ($n = 10$), whose age is unknown[8].

For each analysis day, the number of detections per ID is obtained, excluding the mentioned IDs. Additionally, I discarded all detections with an ID frequency below the 99th percentile of negative IDs. A list of valid IDs per day is kept to filter out wrong detections beforehand.

Implications

[TODO: redo]
The confidence level is set to 95%. This a good balance between gaps in the time series and quality of the data and amount of remaining data. Because bee time series contain a lot of short gaps (mean = 3, 95% confidence), the inference of edges (bees that are spatially close to each other at the same time), should take this into account.

[5]Data set: 26.07.2016, 16:00-16:05
[6]Data set: 20.08.2016, 24 hours, number of total frames: 302400
[7]Probably a mistake in the table, which reports the hatching days for each bee.
[8]id= [2, 74, 2045, 3172, 3764, 3796, 3827, 3836, 3844, 3940]

3.1.2 Defining the Network Pipeline and its Parameters

Pipeline

decisions regarding (2):
weighted/unweighted, directed/undirected
weights: frequency and duration
type: spatial, contact, food
temporal: time-aggregated, time-ordered
decision, because of XY (chriteria?)

The following part describes the pipeline for generating spatial proximity networks out of honey bee tracking data. A node in the network is a bee. They are distinguished by IDs. Only bees are in the network who interact at least once with another bee.

undirected and weighted, aggregated networks

Two bees are associated (spatially close to each other), if their distance is minor to a *maximum distance*. As everything is very close in a bee hive this value is hard to choose. Only this criteria is very week, meaning having a resolution of three frames per seconds results in interactions which could only last for 0.33 seconds. So an additional parameter the *minimum contact duration* is introduced, it is the minimum time they have to spend at least nearby to be called associated.

Taking the fragmentation of tracks into account, it is obvious that two bees could be nearby but not at the very same time, but slightly shifted. So the minimum contact duration would be too errow prone. To overcome this issue one could correct the bee tracks, by filling gaps of varius sizes and interpolating the position of that bee accordingly. This is rather time consuming for this amount of tracking data (TODO: naja so doll auch nicht) and also considering, that the tracking data is going to be improved in the future, then manipulating the raw data seems senseless. I rather perform a gap filling (maybe similar to binary dilation?) on the time series of pairs, but not on the bee tracks, because this is independent of the input data.

Edges are attributed with two parameters. The first one is the frequency of contacts, so how often they share a close position. The second parameter is the total duration of contact, how many time frames in total they spend close by.

The network pipeline takes as input a path to the bb-binary data and outputs a graph in graphML file format. The pipeline takes the following parameters:

- path to data
- confidence in percent
- gap size in frames - this is used to corret the time series of bee pairs
- maximum distance in px - define what close means (spatial proximity)

- minimum contact duration in frames - how many frames bees need to spend nearby
- cutoff in percent - IDs with a number of total detections below X percent of the mean frequency are discarded
- start timestamp - start of network slice
- window size in minutes - size of time window for aggregating the network
- number of used CPUs for parallelization
- year - calculate IDs and set camera setup for 2015 or 2016

The pipeline is parallelized on frame level, that means, each process gets a portion (frames for a timeinterval of five minutes) of the data and extracts interactions/edges. The main process adds everything up and creates a network. The steps are the following:

1. **Filter detections by confidence**
   For each of the four camera the detections are filtered by the confidence level.
2. **Simple stitching**
   Each side of the hive consists of two cameras. The x-coordinates of each detection (of the right cameras) is moved further to the right, also adding an offset of $2 \times$ `maximum distance`. So the left and the right detection of each side of the hive are move into one reference system.
3. **Syncronize Cameras**
   For each side of the hive the cameras need to be syncronized. In the normal case the difference between consecutive frames should be about 0.332 seconds, due to technical problem this value can be lower (0.003) and higher (2.932) at certain times. Cameras 3 and 2 and cameras 1 and 0 are matched, frames without a match are dropped (shorter number of frames, matchen, threshold 0.33/2, minimum).
4. **Discard Detections with certain IDs**
   All detections whos ID is in a list are keept, other detections are discarded. (see frequency filter)
5. **Extract close pairs**
   For each side of the hive, all close pairs according to the maximum distance parameter are calculated and then joined together.
6. **Generate time series of bee pairs**
   The data structure (frames and detection) is transformed to time series of bee pairs.
7. **Correct pair time series.**
   The time series of bees are corrected by filling in the gaps of length `gap size`.
8. **Extract edges**
   The edges and its attributes (frequency and duration) are extracted from the time series of bees using the minimum contact duration parameter. A sequence

of at least X ones counts as one interaction. The frequency of those series adn the total duration (number of ones) are the attributes.

Pipeline Parameters

weighted temporal (time-aggregated) spatial proximity interaction networks out of tracking data
1) parameters for resulting networks: minimum contact duration, start-timestamp, window size in min, maximal distance
2) parameters resulted from dataset characteristics: confidence, file with valid ID lists, gap size

[TODO überarbeiten]

For performing the network analysis, I chose the pipeline parameters as follows:

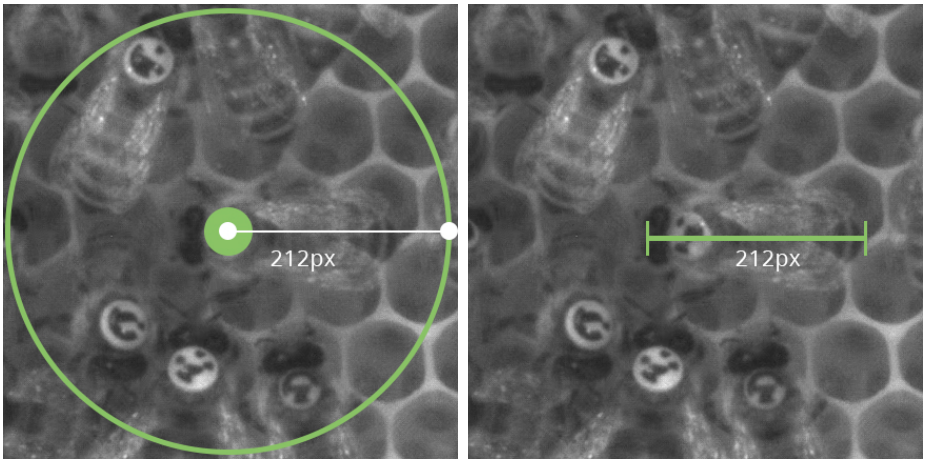
Confidence As explained in section3.1.1, the confidence is set to 95%.

Maximum Distance I chose the length of a bee body, according to Baracchi and Cini [3], as the maximum distance between two bees (figure 3.6a). The average bee length of 212px (±16px) was determinded by manually measuring the length of all bees ($n = 337$) in four images (one for each camera, 21.07.2016, 03:00PM) using the tool ImageJ[9].

Gap Size The gap size is set to two frames. This value corresponds to the median gap length in the time series of pairs (`mode` = 1, `mean` = 27). [TODO: what dataset was used (95% confidence, XXX% cutOff, XXXpx maximal distance, date, camera)]

Minimum Contact Duration This is set to three frames (one second). This corresponds to Mersch et al. [20], they as well exclude interactions below one second. Looking at the frequency distribution of chains of ones (1, 11, 111, and so on) of the pair time series (after filling the gaps), then: `mode` = 1, `median` = 2 and `mean` = 4. Three frames corresponds to 57% of all chains, this seem to be reasonable. [TODO: what dataset was used (95% confidence, XXX% cutOff, XXXpx maximal distance, date, camera)]

[9] `http://imagej.net/Welcome`; Last accessed: 22.02.2016



(a) Contact Radius **(b)** Bee Length

Figure 3.6: Distance Between Bees: A length of a bee is chosen as the maximal distance between bees.

3.1.3 Summary and Results

The goal, as mentioned in 1.2, was to answer the question whether it is possible to infer temporal networks with the provided honey bee tracking data and to work out challenges and limitations regarding the provided data set. Furthermore, it was a goal to identify the parameters necessary for the pipeline.

Parameters

This analysis results in two types of pipeline parameters. The first category specifies the resulting network, concerning the definition of spatial proximity, duration of interaction and size of the aggregated time window. The second type represents parameters resulting out of the characteristics of the dataset.

1. **Network parameters**
   maximum distance, minimum contact duration, window size
2. **Data parameters**
   confidence, list of valid IDs, gap size

Limitations

It is possible to infer networks, but a complex preprocessing of the dataset is essential with two major steps:

1. **Reduction of data**
   Reduce the amount of data to obtain a reliable data set, by filtering out detections with a low confidence value or by IDs with a low detection frequency.
2. **Combine camera data**
   This step consist of the time synchronization of each of the two cameras and the joining of the data per frame.

A tradeoff between the remaining amount of data that can be used for network inference and the data's quality had to be found. A high confidence value reduces the amount of data and produces gaps, whereas the gap size parameter tries to fix this problem.

It is also possible to infer time-aggregated networks, but with restrictions. When limiting the window size for network aggregation to the biological rhythms of day and night[10], then due to a large number of interruptions, only a small amount of useful analysis days remain.

[10] Any other window size entails the inclusion of the duration of biological processes related to honey bees, I would need to know beforehand. Alternatively, I would need to apply a method to infer an appropriate window size out of the given data, this it out of scope. [TODO: ref paper]

Table 3.1: Measures used for analysis Each measure is explained in chapter 2.1.2

Global level measures	Node level measures
Number of nodes N and edges E	Degree k
Average degree $\langle k \rangle$	Strength s
Average strength $\langle s \rangle$	Local clustering coefficient c
Density D	Closeness Centrality C_C
Diameter d_{max}	Betweenness Centrality C_B
Number of components	
Global clustering coefficient C_Δ	
Average path length $\langle d \rangle$	
Edge weights	

3.2 Methods for Analyzing Spatial Proximity Networks

[TODO überarbeiten]
This section explains the what measures I used to investigate the properties of my temporal networks and justifies my choice. Also I explain how I chose a community detection algorithm and which one I picked. Explains method to examine age and spatial segregation of communities and how I study the development of communities.

3.2.1 Investigating the Topology and Network Characteristics

[TODO: überarbeiten]
Table A.2 (or figure A.1 summarized the used network analysis methods in the reviewed studies mentioned in chapter 2.2. The table includes global level measures, node level measures and other network analysis methods the authors used in their studies. I chose the measures for my own analysis, because of XY. [TODO: do I need to explain, why I used this and not that?] Therefore, I am going to analyse the global network properties and local node level properties listed in table 3.1. The node level metrics are investigated also in relation to the bees age. The global network properties are compared to an Erdos-Renyi random network, by averaging over 100 runs [TODO cite?].

3.2.2 Detecting Communities

[TODO: überarbeiten]
(1) check reviwed studies, (2) check comparative analysis, (3) check algos by myselfe. The reviewed studies only include two examples of community and cluster analysis.

Mersch et al. [20] used the infomap [34, 33] algorithm. As they explain this algorithm only works for sparse networks, it is not applicable in my case. Baracchi and Cini [3] use a clustering algorithm. [TODO explain and why not want to use] I want to perform community detection insted of cluster analysis. [TODO: difference?] There are comparative analysis of community detection algorithms, e.g. [41, 16]. They seem to be promising, but assume eighter a power law degree distribution or evaluate networks with a low density, which is not applicable here.

Thererfor, I tested all community detection algorithms implemented in python, to find an algorithm, which works well for my case of animal social networks. The three most common python libraries for network analysis were reviewed: NetworkX[11], igraph[12], and graph-tool[13])

The algorithm needs to fulfill the following criteria:

- Support for large and very dense networks ($N > 1000$, $D > 50$ %)
- Support weighted edges
- Fast runtime

Table 3.2 gives an overview about the twelve algorithms reviewed. Five algorithms did not terminate after 15 minutes and were therefore excluded from further investigations. Infomap and label propagation tend to partition all nodes into a single community, this is known especially in dense graphs [41, 14]. The Louvain algorithm is the same as multilevel, but takes longer producing almost the same communities and therefore was also excluded. Walktrap was tested for different step size parameters, as suggested in [31], the communities remained almost the same (only a few nodes switched communities).

I had a closer look at fastgreedy, leading eigenvector, multilevel, and walktrap regarding the number of detected communities and community size for all three networks. Table 3.3 shows the results. All algorithms found at least two communities. Except for leading eigenvector, there is a tendency that a third community exists. I decided to use two algorithms for community detection: leading eigenvector and walktrap. Farine and Whitehead [11] explains that leading eigenvector is often used with animal social networks and works well. Walktrap is chosen for also examining the possible third community.

Age Distribution of Communities [TODO überarbeiten]
For each community I investigated the age distribution and the average age for. I also investigated whether the age division persists in each snapshot. A two sample Kolmogorov-Smirnov test was used to determine the statistically difference of the age distribution between communities. Answer the question: Communities reflect different age groups? For hypothesis (2) the data is stored as a csv file of birth dates of each bee. For testing if age goups are different the Kolmogorov Smirnov Test was

[11] `https://networkx.github.io/`; Last accessed: 16.03.2016, 6:36 p.m.
[12] `http://igraph.org/python/`; Last accessed: 16.03.2016, 6:38 p.m.
[13] `https://graph-tool.skewed.de/`; Last accessed: 16:03.2016, 6:39 p.m.

Table 3.2: Comparing community detection algorithms Comparison of algorithms implemented in python. Criterias are the support of weighted edges, runtime and number of communities. A runtime indicated by − mean no termination after 15 minutes.

	fastgreedy[1]	leading eigenvector[1]	louvain[2]	multilevel[1]	walktrap[1]	infomap[1]	label propagation[1]	edge betweenness[1]	k-clique communities[2]	optimal modularity[1]	spinglass[1]	statistical inference[3]
Edge weights	×	×	×	×	×	×	×		×	×	×	
Runtime in sec	3.6	6.3	11.7	0.7	19.4	13.2	0.2	–	–	–	–	–
Communities	3	2	2	3	2	1	1	–	–	–	–	–
	473 434 15	488 434	469 453	462 427 33	490 431 (1)	922	922					

[1] igraph, [2] NetworkX, [3] graph-tool

Table 3.3: X X

	fastgreedy	leading eigenvector	multilevel	walktrap
Network 1	473 434 15	488 434	462 427 33	490 431 (1)
Network 2	504 467 7	503 475	481 439 58	372 311 294 (1)
Network 3	534 388	537 385	505 415 (2)	310 390 231

used.

Spatial Distribution of Communities [TODO überarbeiten]
Communities occupy different areas of the comb (similar to [3]). Do they stay at the same in each snapshot? Answer the question: Communities reflect groups of bees working in different areas of the hive? The data which was used to test the hypothesis (1) is saved in a sqlite database for faster access, because using bb_binary (parsing the data over and over again) was to slow.

3.2.3 Evolving Communities

According to Aynaud et al. [1] and Bródka et al. [7] there are three main approaches for community detection in temporal networks (sometimes referred to as community tracking): (1) using a static community detection algorithm on several snapshots and then solving a matching problem, (2) using algorithms that are directly suited for temporal networks and (3) using incremental or online algorithms when processing data streams. For each of the three approaches, several methods already exist. As community tracking is not the main focus of this work, I chose to apply the most natural method out of approach (1): detecting static communities for each snapshot and then matching those communities using set theory.

Two communities at successive time steps are matched if they share enough nodes. The *match value* between two communities C and D according to Hopcroft et al. [17] is defined as:

$$\texttt{match}(C, D) = \texttt{min}\left(\frac{|C \cap D|}{|C|}, \frac{|C \cap D|}{|D|}\right) \tag{3.1}$$

This value is between 0 and 1. A high match value occurs when two communities share many nodes and are of a similar size. Communities with the highest value are matched. The author suggests applying a threshold to more precisely define what "share a lot of nodes" means. Otherwise, a matching could occur between communities with only 0.1% of overlapping nodes. I matched all communities, but excluded values below 3%.

I calculated the match value between consecutive snapshots, to investigate the number of bees, which stay the same over time. Also, I calculated all match values of all communities per snapshot.

3.2.4 Summary

Chapter 4

Results of Network Analysis

This chapter summarizes the analysis results of the temporal, spatial proximity network of honey bees, consisting of three consecutive time-aggregated snapshots. The first section describes results related to static aspects of one snapshot. I investigate the properties of the overall colony and the characteristics of individual bees, concerning its detection frequency and age. Additionally, communities are detected and inspected regarding their practical meaning within the colony. The second section focuses on the temporal network aspects of all three snapshots. I investigated the stability of local and global properties, as well as the stability of functional groups of bees concerning age and spatial distribution. Furthermore, the dynamics of individual bees regarding their group membership over time is examined. The last section fo this chapter summarizes the main results and discusses the findings.

4.1 Static Perspectives of Honey Bee Networks

TODO write some intro

I analyzed a temporal network, consisting of three time-aggregated snapshots; these are referred to below as snapshot 1 ($N = 922$), snapshot 2 ($N = 978$) and snapshot 3 ($N = 922$). The snapshots are aggregated for ten hours (108,000 frames) starting at 8 a.m. and lasting until 6 p.m, see table 4.1 for details about the added bees per day, figure A.10 for the age distributions. Figure 4.1 shows the proportion of intersecting bees between each snapshot. This figure illustrates the stability of the network concerning its size.

Table 4.1: Sampling period Overview of the chosen aggregated daily snapshots including the number of added bees and the time they were added to the hive.

	20.08.16	21.08.16	22.08.16	23.08.16	24.08.16
Snapshot ID	1	-	2	-	3
Number of added bees	0	0	110	60	0
Time added	-	-	2 p.m.	6 p.m.	-

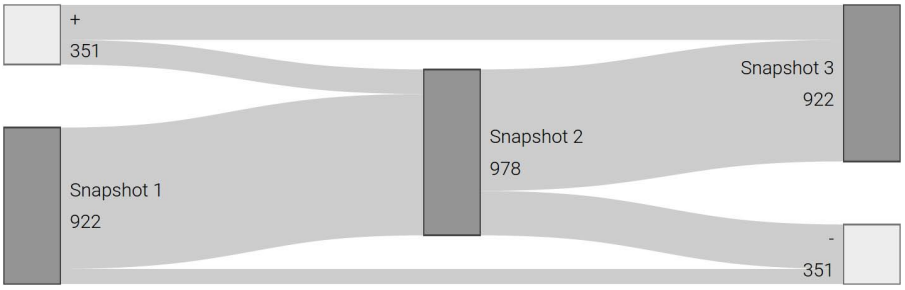


Figure 4.1: Number of bees per snapshot This figure show the amount of bees for each snapshot and the proportion of intersecting bees between snapshots.

4.1.1 Properties of the Colony

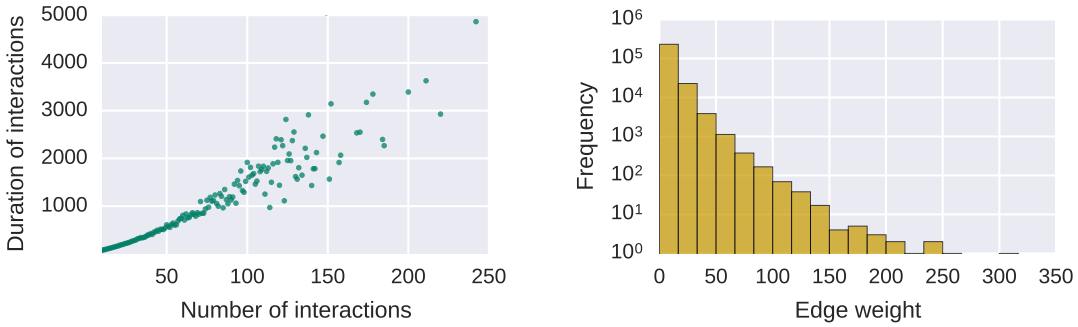
Each snapshot consists of one large component. Table 4.2 summarizes basic network properties. For all, the density D is over 50%. The diameter $\langle d_{\texttt{max}} \rangle$ is three and the average shortest path length $\langle d \rangle$ is below two. The global clustering coefficient C_Δ of all snapshots is higher than compared to an Erdös-Renyi random graph, averaged over 100 runs using the same number of nodes and edges. The high clustering coefficient and the small diameter suggest a small-world network type. On average, each bee is connected to at least 50% of all other bees in the network.

Figure 4.2a shows a positive correlation between the frequency of interactions and the total duration of interactions (averaged). The weight of edges is the frequency of interactions. The edge weight distribution is shown in figure 4.2b. Most edges have a low weight; only a few edges have a high weight. It seems that bees do not prefer individuals bees for interaction.

Table 4.2: Global network properties N is the number of nodes, L the number of edges, D is the diameter, $\langle d_{\texttt{max}} \rangle$ is the average path length, C_Δ the global clustering coefficient, $C_\Delta^{\texttt{rand}}$ is the global clustering coefficient for randomized graph, $\langle k \rangle$ the average degree and $\langle s \rangle$ represents the average strength, as introduced in section 2.1.2.

	N	L	D	$\langle d_{\texttt{max}} \rangle$	$\langle d \rangle$	C_Δ	$\langle k \rangle$	$\langle s \rangle$
Snapshot 1	922	291179	0.69	3	1.32	0.79	631.62	5680.17
Random 1	922	291179	0.69	2	1.31	0.69	631.62	-
Snapshot 2	978	256066	0.54	3	1.46	0.72	523.65	3977.94
Random 2	978	256066	0.54	2	1.46	0.54	523.65	-
Snapshot 3	922	259421	0.61	3	1.39	0.75	562.74	4205.99
Random 3	922	259421	0.61	2	1.39	0.61	562.74	-

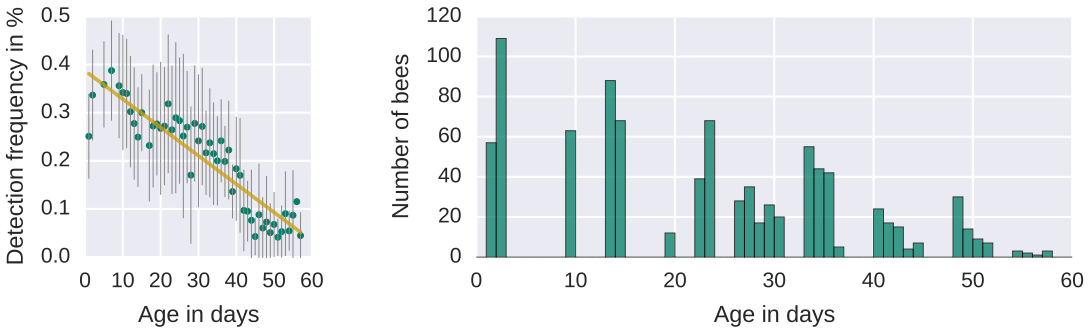
Figure 4.3b shows the age distribution of the investigated snapshot. This distribution does not seem to follow any known distribution. It corresponds to the artificial tagging of bees. Consequently, bees of certain age groups are simply not present. The detection frequency of an individual bee is negatively correlated with its age



(a) Type of edge weights **(b)** Edge weight distribution

Figure 4.2: Edge wights

(figure 4.3a).



(a) **(b)**

Figure 4.3: X

4.1.2 Characteristics of Individual Bees

What is the distribution of values within the colony. Are all bees the same or is there a big difference, maybe indicating two groups to functional groups or different behaviour Question: With how many other bees does a bee interact? Is this the same for all bees, what is the distribution?

The degree of a bee represents the number of other bees this individual interacts with. Figure 4.4a indicates a bimodal degree distribution, with a borderline at about 0.4. This indicates the presence of two groups of bees, one group interacts with less than 40% of the population, and the second group with at least 40% (local peak at 80%). The degree distribution does not follow a power law, but rather a normal distribution. Therefore most bees have the same number of links. This indicates the absence of hubs[1], meaning a few bees beeing highly connected.

Degree and detection frequency are positively correlated, the higher the degree the higher is the detection frequency.

The higher the degree the higher the detection frequency. Bees staying in the hive, are detected more often then for example foragers. Those bees have of course more time to gather links to other bees.

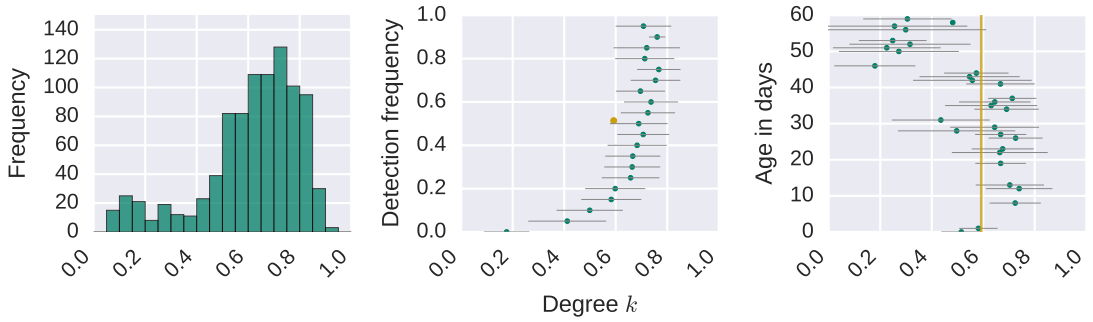
Strength

Degree, Strength and Local Clustering Coefficient and
Betweenness and Closeness Centrality

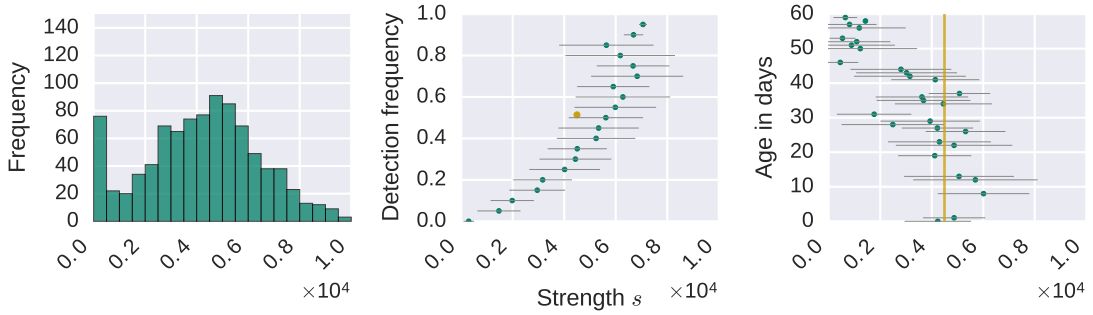
bimodal degree distribution
type of network: no scale free
todo plot in relation to age of bees
todo plot in relation to detection frequency

[TODO]
in relation to age and detection frequency
closeness
betweenness

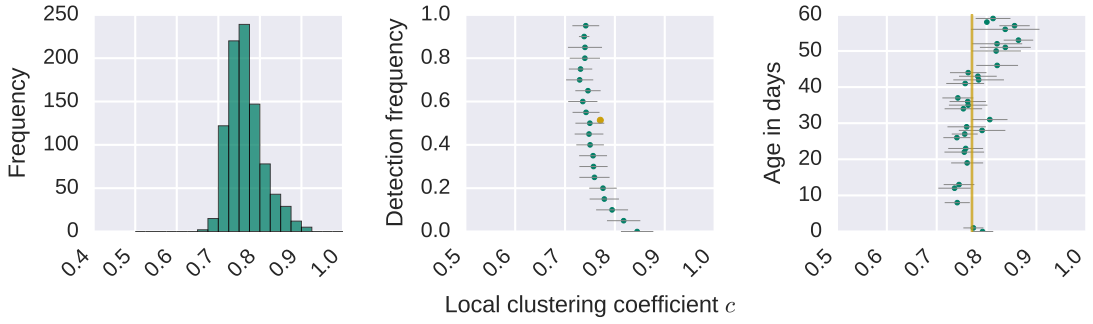
[1]Hubs are highly connected nodes having a large number of links.



(a) Degree

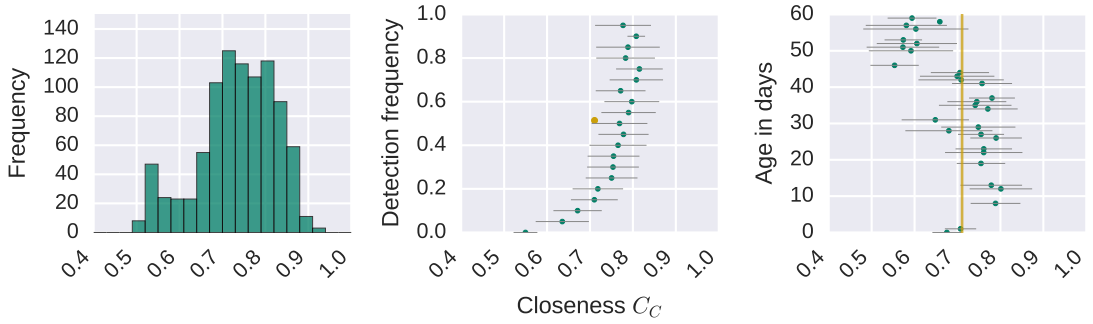


(b) Strength

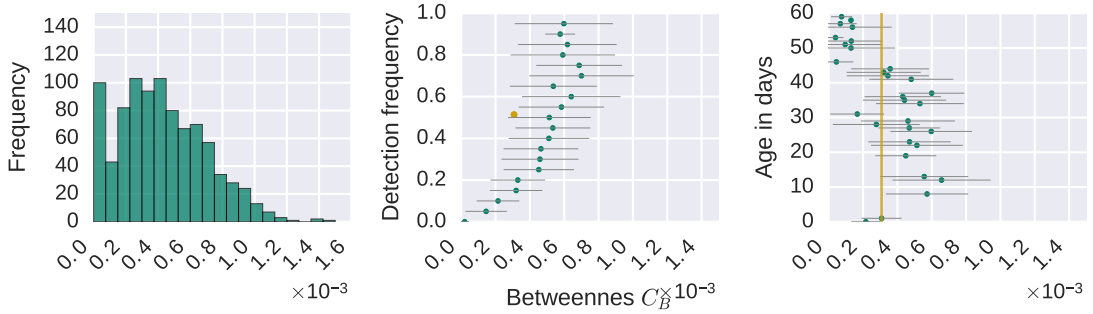


(c) Local clustering coefficient

Figure 4.4: Degree, strength and local clustering coefficient (LCC) in relation to age and detection frequency xxx



(a) Closeness Centrality



(b) Betweeness Centrality

Figure 4.5: Centrality in relation to age and detection frequency xxx

Table 4.3: Communities per algorithm Communities marked with * contain the queen. Age and standard deviation (SD) are measured in days. The queen and bees with a negative age (10 bees).

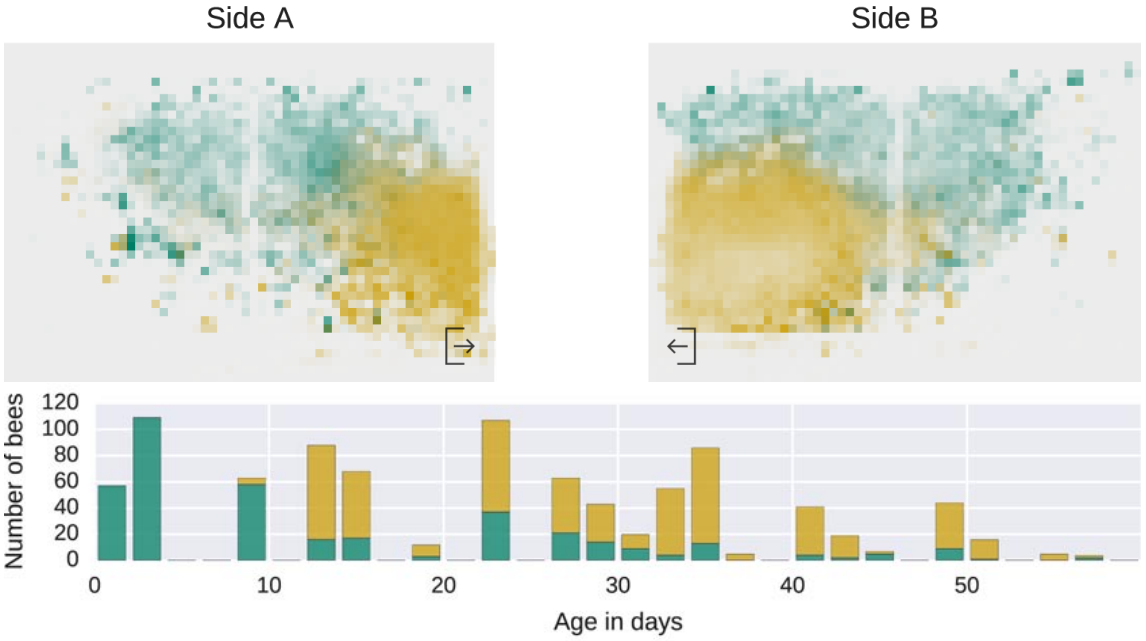
	Community ID	Members	Proportion	Age	SD
LE	CY	∗381	41.78%	13.15	±13.50
	CO	531	58.22%	28.70	±11.67
WT	CY	∗229	25.11%	6.55	±10.36
	CM	298	32.68%	25.08	±11.97
	CO	385	42.21%	29.29	±11.44

Table 4.4: Kolmogorov-Smirnov test for snapshot 3 p-values for leading eigenvector (LE) and walktrap (WT)

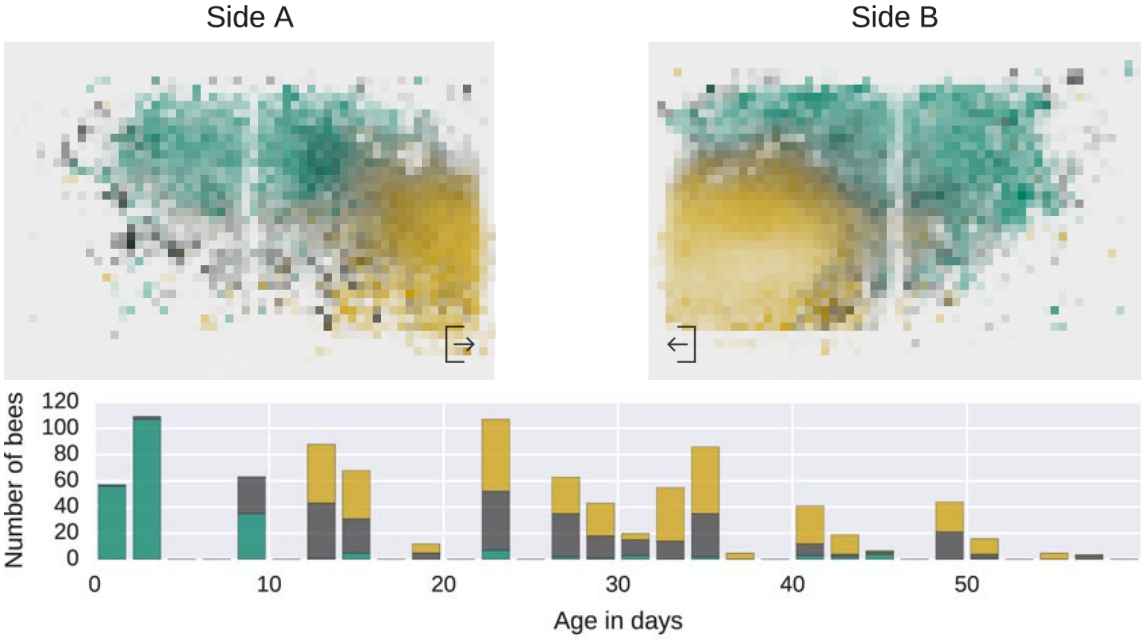
		LE p-value	WT p-value
Network 3	CY3, CO3	5.10e-66	5.51e-67
	CY3, CM3		1.10e-95
	CM3, CO3		1.98e-05

4.1.3 Functional Groups within the Colony

The leading eigenvector community detection algorithms revealed two communities, about the same size, the walktrap algorithm instead three communities (see table 4.3). The communities correspond to separate age groups and are located in different regions on the comb (see figure 4.6). The younger communities are situated in the center and the old communities closer to the hive exit. The middle-aged community (only for walktrap) is located between and on the periphery. Table 4.4 shows the p-values for the two sampel KS-test.



(a) Leading eigenvector communities



(b) Walktrap communities

Figure 4.6: Age and spatial distribution of communities *Green* represents the young community occupying the center are of the comb and *orange* the old community, which is situated closer to the hive access. For walktrap the middle-aged community is depicted in *gray* and is located inbetween.

4.2 Temporal Perspectives of Honey Bee Networks

TODO: zusammen fassen das alles gleich ist!
nur degree und closeness etwas mehr beschreiben.
The edge weight distribution is the same for all three snapshots (figure A.9d). The same is true for the correlation between the age of bees and the frequency this bee was detected (figure).

same degree distribution, can be seen in figure X
same strength distribution and same local clustering coefficient distribution (figure are in appendix)
figure degee distribution
maybe: relation to age and detection frequency also in appendix

same centrality distribution betweenness is in appendix
figure for closeness distribution
maybe relation to age and detection frequency also in appendix

4.2.1 Stability of Functional Groups

Table 4.5 lists the exact number of bees per community for each algorithm and snapshot. For each snapshot, the leading eigenvector detected two communities with about the same number of bees. The first communities CY(1,2,3) contain the queen and on average younger bees than the second communities CO(1,2,3).
In comparison, walktrap identified three communities, but two for the first snapshot. Again the first communities CY(1,2,3) consist of the queen and on average younger bees than the second CM(2,3) and third communities CO(1,2,3). The bees in CM2 and CM3 are on average younger than the bees in CO2 and CO3. Figure A.8 depicts the age distribution for each community and snapshot.

A two-sample Kolmogorov–Smirnov test showed that the age distributions are significantly different ($p < 0.001$) for both algorithms. However, the p-values for the walktrap communities CM2, CO2, and CM3, CO3 are lower.

CY(1,2,3) are located in the center of the comb, CO(1,2,3) closer to the hive access and CM(2,3) are situated in between. This spatial segregation of communities is similar in all three snapshots. For further reference see heat maps in A.7 and A.6.

Functional groups of honey bees seem to differ in their respective age and occupy different areas of the comb.

4.2.2 Dynamic of Individual Bees

Figure 4.7a (leading eigenvector) and figure 4.7b (walktrap) show the flow of community members between consecutive snapshots. For leading eigenvector communities,

Table 4.5: Overview about communities per network Communities marked with * contain the queen. Age and standard deviation (SD) are measured in days. For each network the queen and bees with a negative agre are excluded: network 1 - 12 bees, network 2 - 119 bees, network 3 - 10 bees.

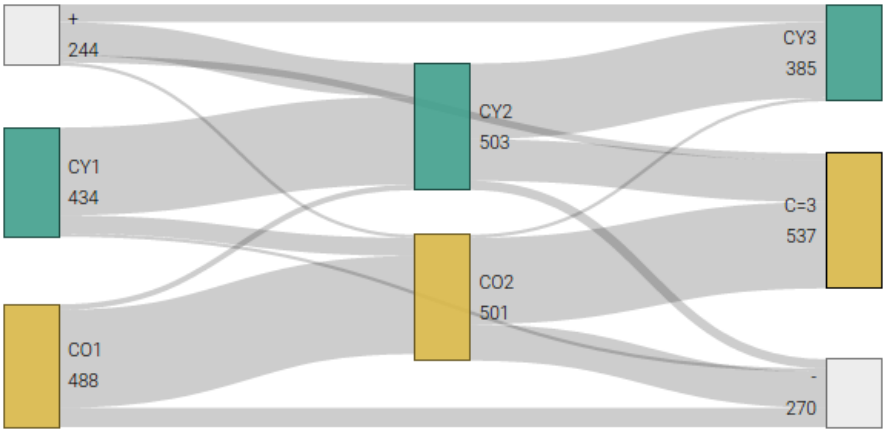
	ID	Members	Proportion	Age	SD
Leading eigenvector					
Network 1	CY1	∗430	47.25%	17.12	±10.97
	CO1	480	52.75%	27.24	±10.96
Network 2	CY2	∗392	45.63%	20.24	±12.01
	CO2	467	54.37%	28.10	±10.88
Network 3	CY3	∗381	41.78%	13.15	±13.50
	CO3	531	58.22%	28.70	±11.67
Walktrap					
Network 1	CY1	∗427	46.92%	17.07	±10.92
	CO1	482	52.97%	27.23	±11.00
Network 2	CY2	∗263	30.62%	18.23	±11.46
	CM2	305	35.51%	25.20	±11.47
	CO2	291	33.88%	29.47	±10.06
Network 3	CY3	∗229	25.11%	6.55	±10.36
	CM3	298	32.68%	25.08	±11.97
	CO3	385	42.21%	29.29	±11.44

Table 4.6: Kolmogorov-Smirnov test p-values for leading eigenvector (LE) and walktrap (WT) for each network and its communities.

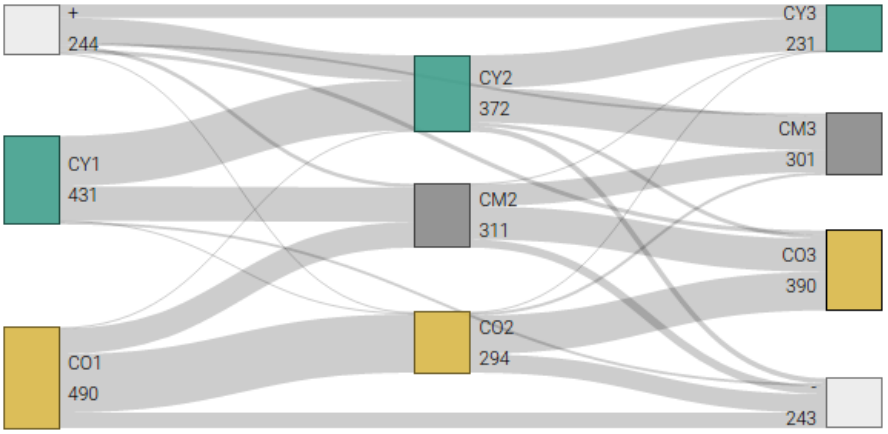
		LE p-value	WT p-value
Network 1	CY1, CO1	2.18e-33	1.52e-32
Network 2	CY2, CO2	2.99e-20	2.3e-32
	CY2, CM2		4.72e-10
	CM2, CO2		1.00e-04
Network 3	CY3, CO3	5.10e-66	5.51e-67
	CY3, CM3		1.10e-95
	CM3, CO3		1.98e-05

the majority of the bees stay in their age group, and a small fraction of bees switches to older communities. Only a few bees change to younger communities. The new middle-aged communities CM2 and CM3, detected by walktrap, consist partly of young (CY1) and old (CO1) bees. The switching behavior of individuals between communities is similar to leading eigenvector.

Individual bees change communities as they age.



(a) Leading eigenvector communities



(b) Walktrap communities

Figure 4.7: Dynamic community members Each column represents a time step, the colored rectangles represent the communities for each time step, and the height of the rectangles corresponds to the number of its community members, as referenced by the number. *Green* indicates the community containing young bees and the queen, *gray* represents the community containing middle-aged bees (only for walktrap), and *orange* the community containing old bees. This figure shows that the major part of the members eighter stay in the same aged community or switch to an older group.

4.3 Discussion of Results

stable global structure over time
degree dist, centrality and so on stays the same over the three snapshots
dynamic local structure (node level, individual bee)
bees change communities as they age
conclusion: verifies my definition of the networks and chosen parameters

Chapter 5

Conclusion and Future Work

what your findings might mean, how valuable they are and why
what was the purpose of the study
summarize the approach
major findings: summarize the results

relates directly to the research question and objectives
contribution to the knowledge
significance of the study
maybe state a personal opition

5.1 Limitations

methods
implementation
tagging, dataset
context, interdisziplinary

5.2 Recommendations

recommendations for further studies
recommendation for change
each recommendation should directly trace a direct conclusion

Bibliography

[1] Thomas Aynaud et al. “Communities in evolving networks: definitions, detection, and analysis techniques”. In: *Dynamics On and Of Complex Networks, Volume 2*. Springer, 2013, pp. 159–200.

[2] Albert-László Barabási. *Network science*. Cambridge University Press, 2016.

[3] David Baracchi and Alessandro Cini. “A Socio-Spatial Combined Approach Confirms a Highly Compartmentalised Structure in Honeybees”. In: *Ethology* 120.12 (2014), pp. 1167–1176.

[4] Alain Barrat et al. “The architecture of complex weighted networks”. In: *Proceedings of the National Academy of Sciences of the United States of America* 101.11 (2004), pp. 3747–3752.

[5] Benjamin Blonder and Anna Dornhaus. “Time-ordered networks reveal limitations to information flow in ant colonies”. In: *PloS one* 6.5 (2011), e20298.

[6] Benjamin Blonder et al. “Temporal dynamics and network analysis”. In: *Methods in Ecology and Evolution* 3.6 (2012), pp. 958–972.

[7] Piotr Bródka, Stanisław Saganowski, and Przemysław Kazienko. “Community Evolution”. In: *Encyclopedia of Social Network Analysis and Mining* (2014), pp. 220–232.

[8] Daniel Charbonneau, Benjamin Blonder, and Anna Dornhaus. “Social insects: a model system for network dynamics”. In: *Temporal Networks*. Springer, 2013, pp. 217–244.

[9] James D Crall et al. “BEEtag: a low-cost, image-based tracking system for the study of animal behavior and locomotion”. In: *PloS one* 10.9 (2015), e0136487.

[10] Darren P Croft, Richard James, and Jens Krause. *Exploring animal social networks*. Princeton University Press, 2008.

[11] Damien R Farine and Hal Whitehead. “Constructing, conducting and interpreting animal social network analysis”. In: *Journal of Animal Ecology* 84.5 (2015), pp. 1144–1163.

[12] Mark Fiala. “Comparing artag and artoolkit plus fiducial marker systems”. In: *Haptic Audio Visual Environments and their Applications, 2005. IEEE International Workshop on*. IEEE. 2005, 6–pp.

[13] Vincent A Formica et al. “Fitness consequences of social network position in a wild population of forked fungus beetles (Bolitotherus cornutus)”. In: *Journal of evolutionary biology* 25.1 (2012), pp. 130–137.

[14] Santo Fortunato. “Community detection in graphs”. In: *Physics reports* 486.3 (2010), pp. 75–174.

[15] Efrat Greenwald, Enrico Segre, and Ofer Feinerman. “Ant trophallactic networks: simultaneous measurement of interaction patterns and food dissemination”. In: *Scientific reports* 5 (2015).

[16] Steve Harenberg et al. “Community detection in large-scale networks: a survey and empirical evaluation”. In: *Wiley Interdisciplinary Reviews: Computational Statistics* 6.6 (2014), pp. 426–439.

[17] John Hopcroft et al. “Tracking evolving communities in large linked networks”. In: *Proceedings of the National Academy of Sciences* 101.suppl 1 (2004), pp. 5249–5253.

[18] Raphaël Jeanson. “Long-term dynamics in proximity networks in ants”. In: *Animal Behaviour* 83.4 (2012), pp. 915–923.

[19] Jens Krause et al. *Animal social networks*. Oxford University Press, USA, 2014.

[20] Danielle P Mersch, Alessandro Crespi, and Laurent Keller. “Tracking individuals shows spatial fidelity is a key regulator of ant social organization”. In: *Science* 340.6136 (2013), pp. 1090–1093.

[21] James Moody, Daniel McFarland, and Skye Bender-deMoll. “Dynamic network visualization 1”. In: *American journal of sociology* 110.4 (2005), pp. 1206–1241.

[22] Dhruba Naug. “Structure and resilience of the social network in an insect colony as a function of colony size”. In: *Behavioral Ecology and Sociobiology* 63.7 (2009), pp. 1023–1028.

[23] Dhruba Naug. “Structure of the social network and its influence on transmission dynamics in a honeybee colony”. In: *Behavioral Ecology and Sociobiology* 62.11 (2008), pp. 1719–1725.

[24] Dhruba Naug and Brian Smith. “Experimentally induced change in infectious period affects transmission dynamics in a social group”. In: *Proceedings of the Royal Society of London B: Biological Sciences* 274.1606 (2007), pp. 61–65.

[25] M. Newman. *Networks: An Introduction*. OUP Oxford, 2010.

[26] Mark EJ Newman. “Finding community structure in networks using the eigenvectors of matrices”. In: *Physical review E* 74.3 (2006), p. 036104.

[27] Michael C Otterstatter and James D Thomson. “Contact networks and transmission of an intestinal pathogen in bumble bee (Bombus impatiens) colonies”. In: *Oecologia* 154.2 (2007), pp. 411–421.

[28] Gergely Palla et al. “Uncovering the overlapping community structure of complex networks in nature and society”. In: *Nature* 435.7043 (2005), pp. 814–818.

[29] Noa Pinter-Wollman et al. “The dynamics of animal social networks: analytical, conceptual, and theoretical advances”. In: *Behavioral Ecology* 25.2 (2014), p. 242. eprint: `/oup/backfile/Content_public/Journal/beheco/25/2/10.1093/beheco/art047/2/art047.pdf`.

[30] Noa Pinter-Wollman et al. “The effect of individual variation on the structure and function of interaction networks in harvester ants”. In: *Journal of the Royal Society Interface* 8.64 (2011), pp. 1562–1573.

[31] Pascal Pons and Matthieu Latapy. “Computing communities in large networks using random walks”. In: *International Symposium on Computer and Information Sciences*. Springer. 2005, pp. 284–293.

[32] Lauren E Quevillon et al. “Social, spatial, and temporal organization in a complex insect society”. In: *Scientific reports* 5 (2015).

[33] Martin Rosvall and Carl T Bergstrom. “An information-theoretic framework for resolving community structure in complex networks”. In: *Proceedings of the National Academy of Sciences* 104.18 (2007), pp. 7327–7331.

[34] Martin Rosvall, Daniel Axelsson, and Carl T Bergstrom. “The map equation”. In: *The European Physical Journal Special Topics* 178.1 (2009), pp. 13–23.

[35] Jacob Scholl and Dhruba Naug. “Olfactory discrimination of age-specific hydrocarbons generates behavioral segregation in a honeybee colony”. In: *Behavioral Ecology and Sociobiology* 65.10 (2011), p. 1967.

[36] Ana B Sendova-Franks et al. “Emergency networking: famine relief in ant colonies”. In: *Animal Behaviour* 79.2 (2010), pp. 473–485.

[37] Leon Sixt. “RenderGAN: Generating realistic labeled data - with an application on decoding bee tags”. B.S. Thesis. Freie Universtät Berlin.

[38] Fernando Wario et al. “Automatic methods for long-term tracking and the detection and decoding of communication dances in honeybees”. In: (2015).

[39] Stanley Wasserman and Katherine Faust. *Social network analysis: Methods and applications*. Vol. 8. Cambridge university press, 1994.

[40] James S Waters and Jennifer H Fewell. “Information processing in social insect networks”. In: *PLoS One* 7.7 (2012), e40337.

[41] Zhao Yang, René Algesheimer, and Claudio J Tessone. “A Comparative Analysis of Community Detection Algorithms on Artificial Networks”. In: *Scientific Reports* 6 (2016).

List of Figures

List of Tables

Appendix A

Appendix Stuff

A.1 Network Analysis

Table A.1: Summary social insect studies https://docs.google.com/spreadsheets/d/1eKuPU-XmqwrHkS_5-TgS8UnO5O-Hwe1kyRIpareywP4/edit?usp=sharing

	TODO	TODO
x	x	x
x	x	x

Table A.2: Network measures of studies https://docs.google.com/spreadsheets/d/1eKuPU-XmqwrHkS_5-TgS8UnO5O-Hwe1kyRIpareywP4/edit?usp=sharing

	TODO	TODO
x	x	x
x	x	x

Table A.3: Network types of studies https://docs.google.com/spreadsheets/d/1eKuPU-XmqwrHkS_5-TgS8UnO5O-Hwe1kyRIpareywP4/edit?usp=sharing

	TODO	TODO
x	x	x
x	x	x

	Temporal Analysis			Static Analysis											
	blonder2011time	jeanson2012long	mersch2013tracking	**naug2007**	otterstatter2007contact	**naug2008structure**	naug2009structure	sendova2010	pinter2011effect	**scholl2011olfactory**	waters2012infomation	**baracchi2014socio**	greenwald2015ant	quevillon2015social	**score**
Global level measures															
Average degree	x										x			x	**3**
Maximal degree											x				1
Average strength					x	x									**2**
Average path length							x				x				2
Density					x		x				x				**3**
Diameter											x				1
Node level measures															
Degree	x						x	x			x			x	**5**
Strength		x			x	x			x			x			**5**
Betweennes centrality	x	x												x	**3**
Closeness centrality		x										x		x	**3**
Eigenvector centrality												x			**1**
Clustering coefficient						x	x								**2**
Other method															
Burst constaint														x	1
Disparity		x													2
Cluster or Community detection			x									x			**2**
Fitting of distributions	x								x		x				**3**
Compare to random						x	x								**2**
Information flow	x					x							x		**2**
Interaction between age groups										x					**1**
Ego network												x			**1**
Robustness						x									**1**

Figure A.1: XXX XXX

	Temporal Analysis			Static Analysis										
	blonder2011time	jeanson2012long	mersch2013tracking	**naug2007**	otterstatter2007contact	**naug2008structure**	naug2009structure	sendova2010	pinter2011effect	**scholl2011olfactory**	waters2012information	**baracchi2014socio**	greenwald2015ant	quevillon2015social
Type of network	ta/to	ta	ta	s	s	s	s	s	s	s	s	s	s	s
Weighted Network														
duration of interaction	-	x	-	-	x	x	-	-	-	-	-	-	v (2)	-
number of interactions	-	-	x	-	x	-	x	-	x	-	-	x		-
Directed Network														
directed	x- (1)	-	-	x	-	x	-	x	-	-	x	-	x	-
Type of interaction														
spatial proximity (body(B) length)		4/3xBL							2/3xBL >0.2s			1xBL		
physical contact	A-B		A		B-B		ex (3)			A	A			
food exchange (throphallaxis)				> 5s		> 5s		x		x			x	> 1s

(1) both
(2) volume corresponds to duration
(3) except dominance interactions

ta = time-aggregates, to=time-ordered, s=static
A = antenna
B = body
BL = bodylength

Figure A.2: XXX XXX

	Temporal Analysis			Static Analysis										
	blonder2011time	jeanson2012long	mersch2013tracking	**naug2007**	otterstatter2007contact	**naug2008structure**	naug2009structure	sendova2010	pinter2011effect	**scholl2011olfactory**	waters2012information	**baracchi2014socio**	greenwald2015ant	quevillon2015social
Tracking														
automatic		x	x		x				x				x	
manual	x			x		x	x	x		x	x	x		x
Species														
	A	A	A	HB	BB	HB	W	A	A	HB	A	HB	A	A
(1) Time														
Total duration of study	3w	3w	41d	1d	40d		24d (6)	1d	1d	3w	1d	1d	1d	8d (5)
Observation period	2x 30m	3x7x 24h	41x 24h	1h	40x 12h (4)	1h	24x 45x5m	30m	5m	3x 1h	2h	10h	30m	8x 30m
Sampling resolution***	v/e	1 f/s	2 f/s	v/e	30 f/s	v/e	v/e	1 f/m	30 f/s	v/e	15 f/s	1 f/m	v/e	v/e
(2) Space*														
1-frame hive												x (3)		
2-frame hive				x (2)		x(4)				x				
(3) Size														
Number of colonies	4	4	6	1	7	1	9	4	2	1	2	1	2(1)	2
Colonie size**	6-90	55-58	122-192	4000	5-7	1000	8-40	42-95	131, 72	1500	89	4000	50-100	75
Marked individuals	x	x	x		x		x	x	x		x	211	x	x
Marked cohorts				6		4				3				
Age			x	x		x				x		x		
Analysis Tools in R														
igraph	x										x			x
t-net		x												
timeordered	x													
Other Tools: netdraw, cytoscape, UCINET, FANMOD														

(1) two species
(2) only video for one side "(entrance desiged so foragers should unload here)
(3) only one side observed
(4) 6 day and 6 night"
(5) night
(6) Each sampling day consisted of three sessions of 2 h each between 0630 and 1830 hours. in each session15 5-min all-occurrence samplings were carried out resulting.

A = Ant
BB = Bumble Bee "HB = Honey bee
W = Wasp"

* only for honey bees
** Mean or range if > 2
**** v=video, e=event, if no resolution given or manual video analysis was used

Figure A.3: XXX XXX

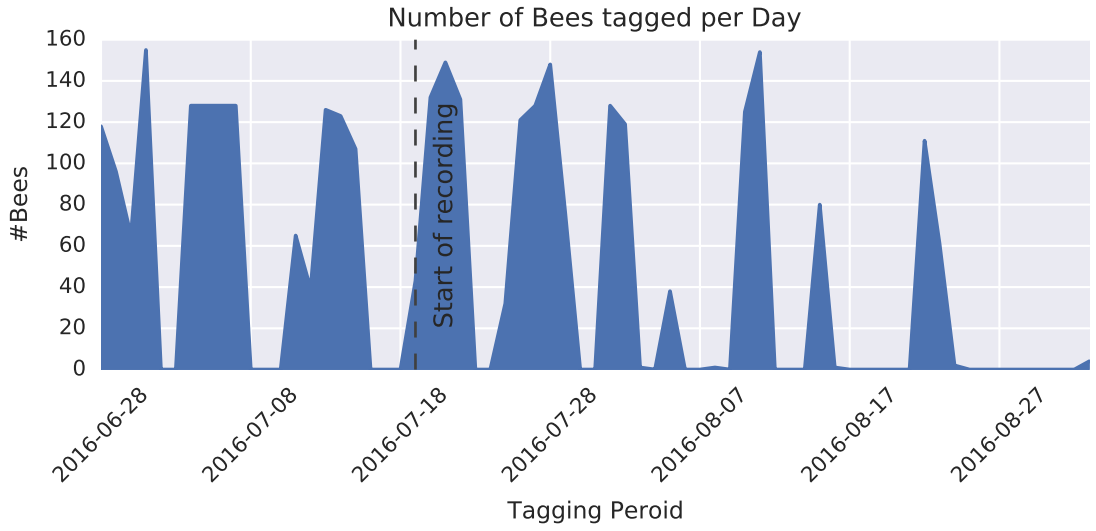


Figure A.4: Tagging frequency The bees were primarily tagged during the week. On average 48 bees were tagged each day, considering only tagging days, the average is about 91. [TODO: combine with other image or make nicer!]

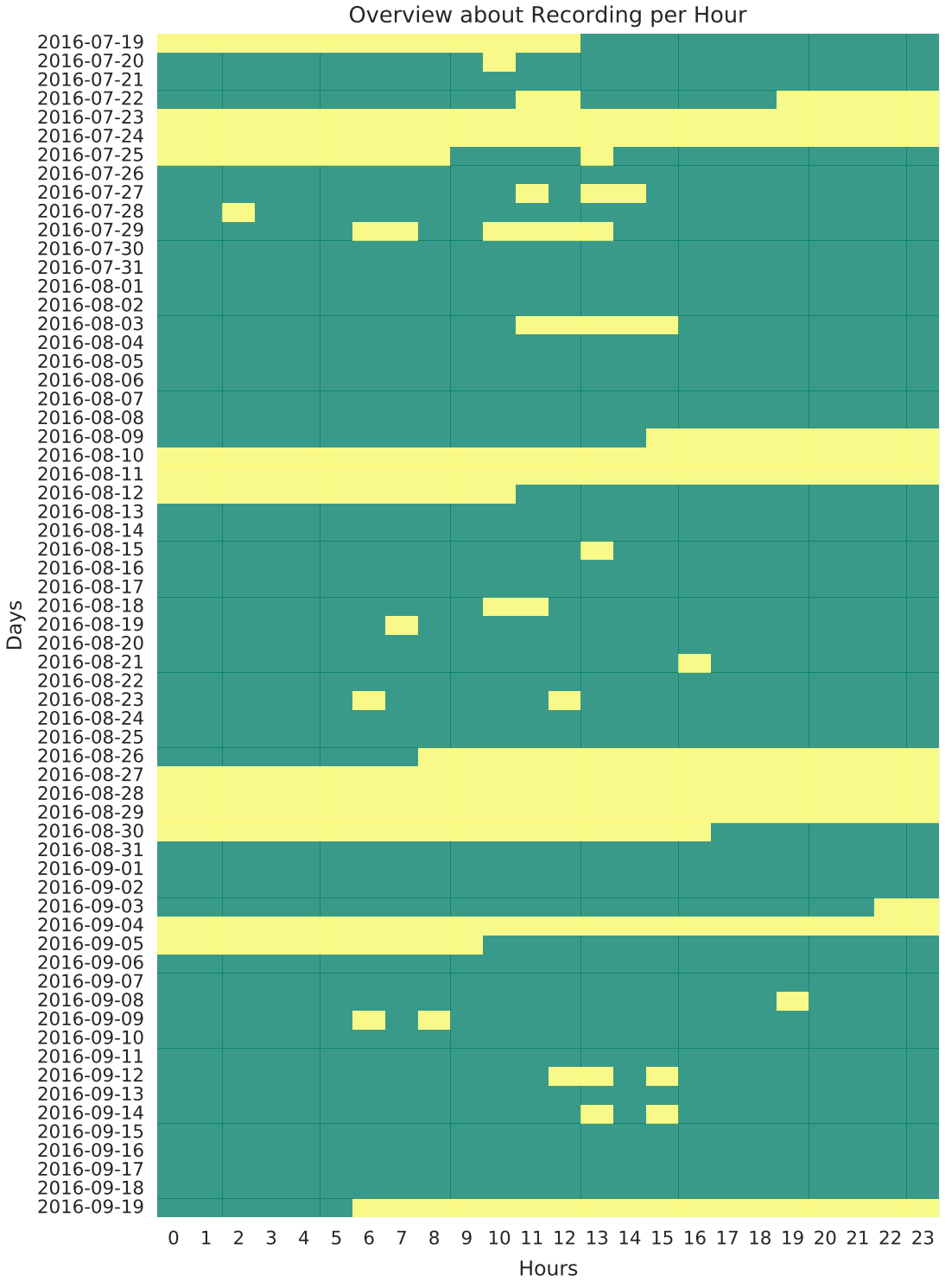
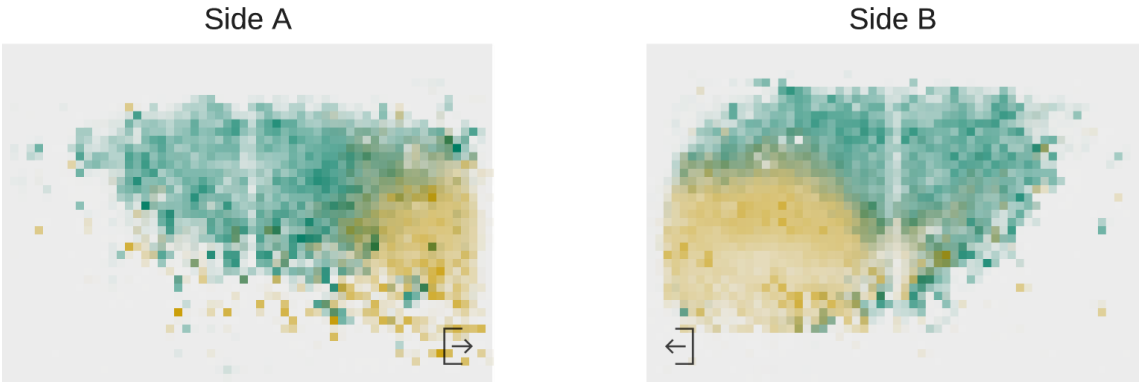
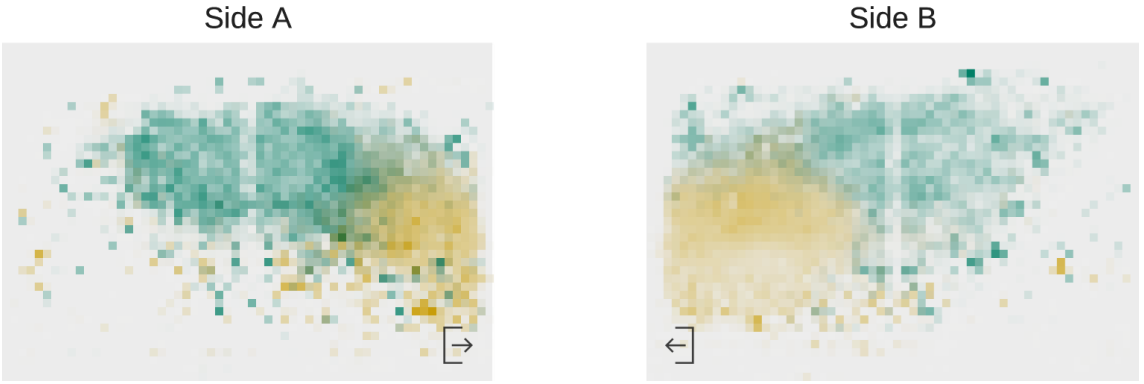


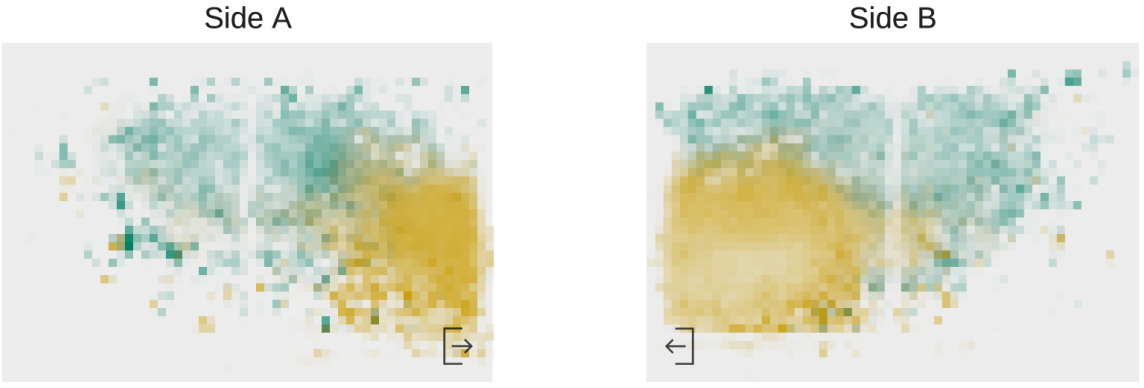
Figure A.5: Recording season with maintainance and failures *Green* indicates recording went without any big interruption; *Yellow* indicates maintainance work or technical failures of one or all cameras. This is calculated using the expected number of files produced by each camera per hour. [TODO, reduzieren auf eine Info pro Tag (keine stuendliche aufloesung), kombinieren mit anzahl der getaggten bienen pro tag, und welchen Zeitraum hab ich nun verwendet], ausserdem Zeit von links nach rechts!, evtl. kein Datum, sonder Tage durchnummerieren



(a) Network 1

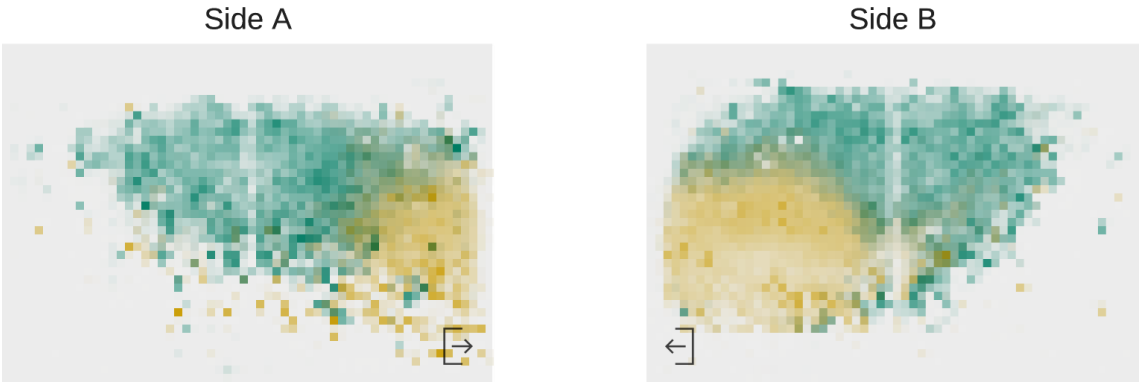


(b) Network 2

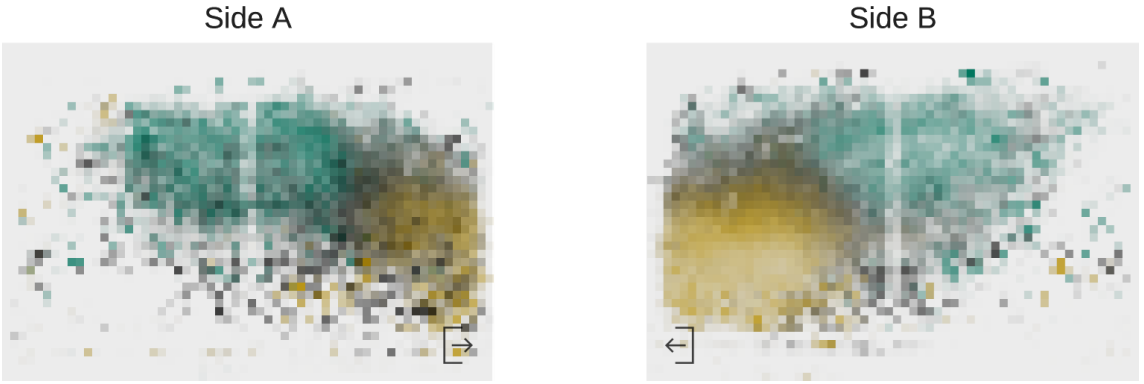


(c) Network 3

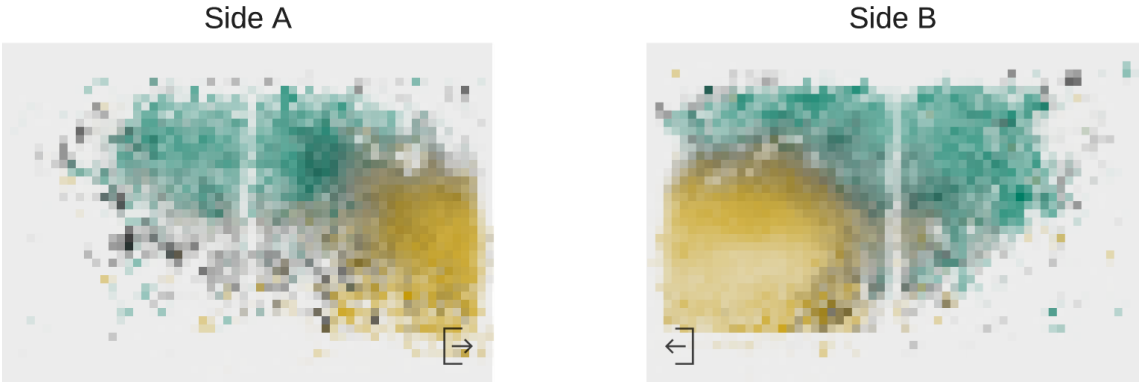
Figure A.6: Communities per network - leading eigenvector The *green* colour represents the younger community, containing the queen. The *orange* color represents the older community. The hive exit on side A is on the bottom right and on side B on the bottom left. The data is aggredated for the complete timeframe of ten hours.



(a) Network 1

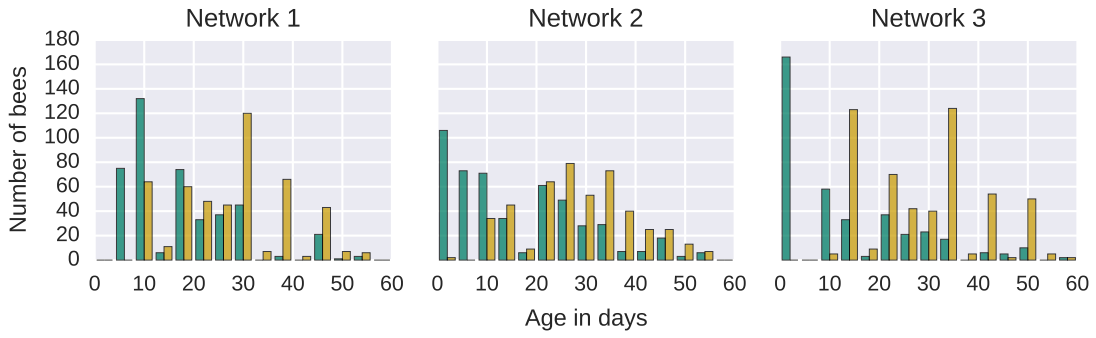


(b) Network 2

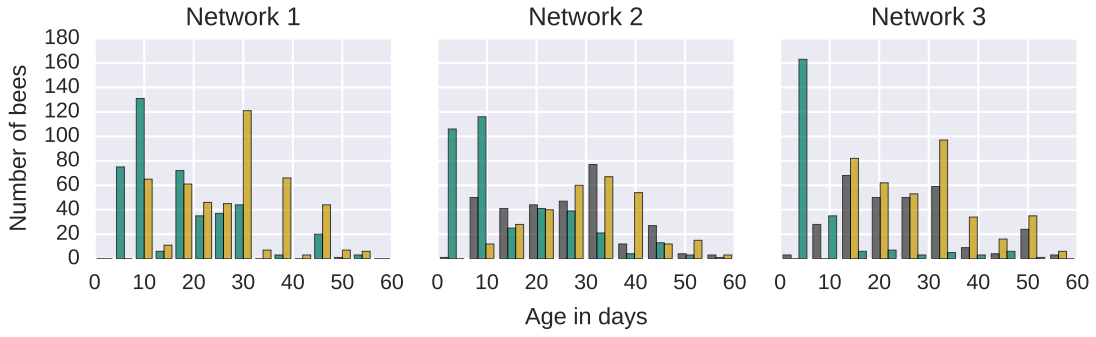


(c) Network 3

Figure A.7: Communities per network - walktrap The *green* colour represents the younger community, containing the queen. The *orange* color represents the older community. The *gray* represents the middle-age community. The hive exit on side A is on the bottom right and on side B on the bottom left. The data is aggredated for the complete timeframe of ten hours.

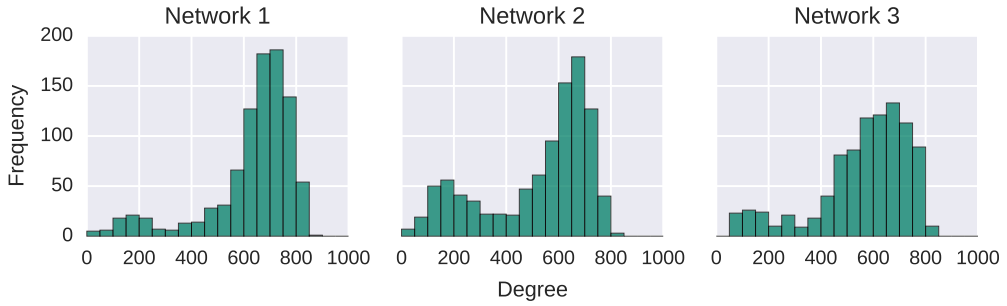


(a) Leading eigenvector

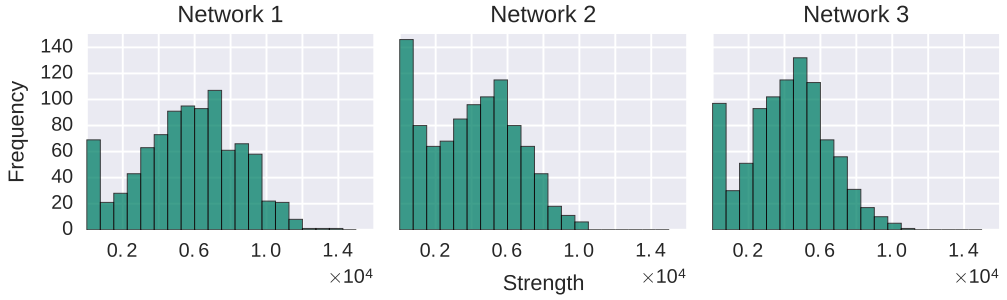


(b) Walktrap

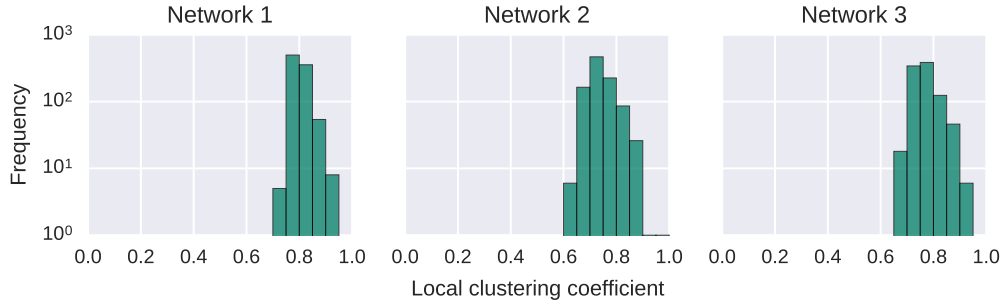
Figure A.8: Age distribution for each community and network The *green* bar is the community containing the queen. The queens age is not included in the statistic. The *orange* bars coresspond to the second community, containing older bees. The *gray* bars is a third community only revealed by walktrap and contains middle-aged bees.



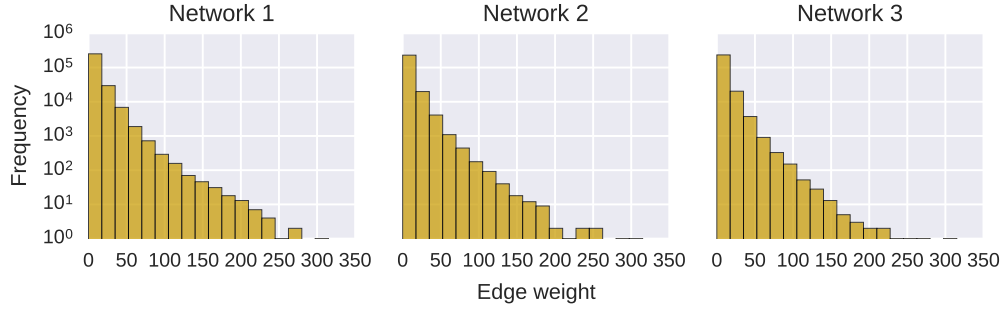
(a) Degree distribution



(b) Strength distribution



(c) Local clustering coefficient



(d) Edge weight distribution

Figure A.9: Degree, strength and edge weight distribution for all three networks.

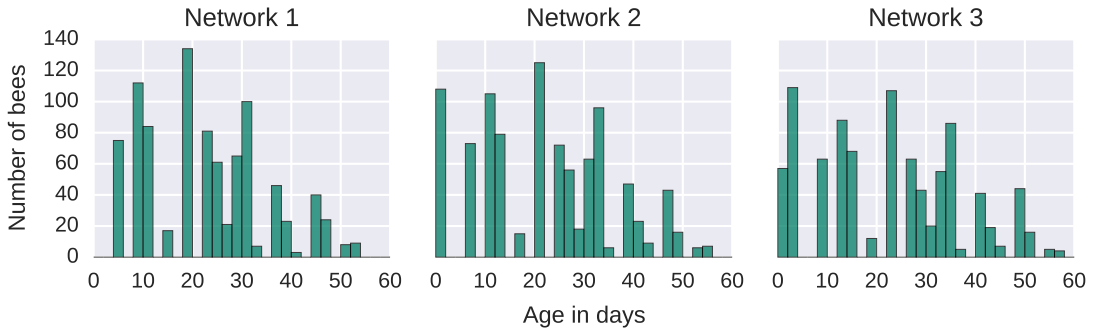


Figure A.10: Age distribution per network The width of a bar corresponds to two days. For each network bees with a negative age and the queen were removed (11, 10, and 9 bees).