

UNIVERSITY OF TWENTE.

Faculty of Electrical Engineering, Mathematics & Computer Science

Inferring Animal Social Interaction Using Proximity Based on BLE and LoRa

Coen Hazekamp M.Sc. Thesis August 2018

> Supervisor: Dr. N. Meratnia

Pervasive Systems Faculty of Electrical Engineering, Mathematics and Computer Science University of Twente P.O. Box 217 7500 AE Enschede The Netherlands

Preface

I received the help from a lot of people in the process of writing this thesis. In the first place from my supervisor, Nirvana Meratnia, for her guidance, insights and feedback. I also received help from the people in the Pervasive Systems group of the University of Twente. Especially Eyuel Debebe Ayele and Jacob Kamminga took the time for giving input and feedback whenever I had questions. I would also like to thank Paul Havinga for being part of the examination committee and of course the rest of the group for giving me a good time whenever I was around, including the moments with smalltalk during breaks.

I should of course also thank my parents and friends. Both the friends that I already know for a lot of years (some even for more than a decade) while living in Enschede, but also the people that I have met during the last two to three years while studying at the University of Twente. They not only helped me relax when I had to but also helped me out a couple of times. I would especially like to thank the people who helped me with gathering data. Even though it was about 34° Celsius outside they were prepared to help me, for which I am really grateful. Among them were not only friends but also family of friends and friends of friends. Before finishing this thesis I asked Lysanne Snijders, a biologist and author of a paper about animal social network analysis, for feedback on my thesis. I want to thank her for reading my thesis, giving feedback and for the quick and enthusiastic replies on my questions.

Some of the aforementioned persons helped more than others, but I appreciate the help of every single one of them.

Abstract

When hearing the term *social network analysis*, most people will think about online social networks like Facebook and Twitter, their targeted advertising models, privacy issues and controversies like the *Cambridge Analytica* scandal. However, human social network analysis is much older than this, originating from the early 1900s with notable advances made in the 1940s, 1950s and 1970s. Technological advances, like the smartphone, made it possible to electronically collect information about human social networks, making the topic more relevant than ever.

Social network analysis is nowadays not only used to infer information about *human* social networks but is also used for the monitoring of the social networks of animals, *animal social network analysis*. Biologists can use this information to analyse animal movement data, study animal behaviour or for the purpose of wildlife conservation. In the latter case, animal social network analysis can for example be used to see if a group or groups of animals are living too close together, leading to aggressive interactions and stress. Based on this information, the conservation biologists can decide to change things in the environment, with the goal to decrease the number of aggressive interactions, decrease stress and monitor what the effect of the environmental changes are. The metrics used in animal social network analysis are *density, community detection, component detection, betweenness* and *degree*.

For the assignment, it is assumed that animal social network information is electronically gathered using tags that collect proximity data and are capable of low-power radio communication using Bluetooth Low Energy (BLE) and Long Range (LoRa), which are known for their low power consumption. Proximity data between animals is currently gathered using GPS or by using proximity sensors. The goal of the assignment is to find out if Bluetooth Low Energy and/or LoRa can be used to collect proximity data. If that is the case, Bluetooth and LoRa might be used to replace GPS and proximity sensors.

This thesis covers the history and background of social network analysis and its applications, animal social network analysis and the used metrics, background information about Bluetooth Low Energy, Long Range, path loss phenomena and shadowing. Using the gathered knowledge about (animal) social network analysis, a simulation model is used, combined with path loss models and shadowing, to simulate the behaviour of BLE and LoRa. Tests with hardware are used for the validation of the results.

Contents

1.1 Background of human social interaction 3 1.2 Indicators of human social interaction 5 1.3 Applications of social network analysis 6 1.4 Gathering animal movement data 8 1.5 Problem statement 9 2 Background 12 2.1 Graph theory 12 2.2 Animal social network data analysis 13 2.2.1 Animal social network metrics 14 2.3 Practical examples using animal social network metrics 15 2.3.1 Other evaluation metrics 17 2.4 Wireless technologies 18 2.4.1 Bluetooth Low Energy Overview 18 2.4.2 Long Range (LoRa) Overview 19 2.4.3 Path loss 20 2.4.4 Path loss 20 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 30 3.2 Experimental design 30 3.3.	1	Intro	oduction	1
1.2 Indicators of human social interaction 5 1.3 Applications of social network analysis 6 1.4 Gathering animal movement data 8 1.5 Problem statement 9 2 Background 12 2.1 Graph theory 12 2.2 Animal social network data analysis 13 2.2.1 Animal social network metrics 14 2.3 Practical examples using animal social network metrics 15 2.3.1 Other evaluation metrics 17 2.4 Wireless technologies 18 2.4.1 Bluetooth Low Energy Overview 18 2.4.2 Long Range (LoRa) Overview 19 2.4.3 Path loss 19 2.4.4 Path loss models 20 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3.1		1.1	Background of human social network analysis	3
1.3 Applications of social network analysis 6 1.4 Gathering animal movement data 8 1.5 Problem statement 9 2 Background 12 2.1 Graph theory 12 2.2 Animal social network data analysis 13 2.2.1 Animal social network data analysis 13 2.2.2 Animal social network data analysis 13 2.2.1 Animal social network data analysis 13 2.2.1 Animal social network metrics 14 2.3 Practical examples using animal social network metrics 15 2.3.1 Other evaluation metrics 17 2.4 Wireless technologies 18 2.4.1 Bluetooth Low Energy Overview 18 2.4.2 Long Range (LoRa) Overview 19 2.4.3 Path loss 19 2.4.4 Path loss models 20 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 <th></th> <th>1.2</th> <th>Indicators of human social interaction</th> <th>5</th>		1.2	Indicators of human social interaction	5
1.4 Gathering animal movement data 8 1.5 Problem statement 9 2 Background 12 2.1 Graph theory 12 2.2 Animal social network data analysis 13 2.2.1 Animal social network metrics 14 2.3 Practical examples using animal social network metrics 15 2.3.1 Other evaluation metrics 17 2.4 Wireless technologies 18 2.4.1 Bluetooth Low Energy Overview 18 2.4.2 Long Range (LoRa) Overview 19 2.4.3 Path loss 19 2.4.4 Path loss models 20 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 37 4.1 Used simulat		1.3	Applications of social network analysis	6
1.5 Problem statement 9 2 Background 12 2.1 Graph theory 12 2.2 Animal social network data analysis 13 2.2.1 Animal social network metrics 14 2.3 Practical examples using animal social network metrics 15 2.3.1 Other evaluation metrics 17 2.4 Wireless technologies 18 2.4.1 Bluetooth Low Energy Overview 18 2.4.2 Long Range (LoRa) Overview 19 2.4.3 Path loss 20 2.4.4 Path loss 20 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 31 3.3 Ivalition datasets 37 4.2 Path loss model setup		1.4	Gathering animal movement data	8
2 Background 12 2.1 Graph theory 12 2.2 Animal social network data analysis 13 2.2.1 Animal social network metrics 14 2.3 Practical examples using animal social network metrics 15 2.3.1 Other evaluation metrics 17 2.4 Wireless technologies 18 2.4.1 Bluetooth Low Energy Overview 18 2.4.2 Long Range (LoRa) Overview 19 2.4.3 Path loss 19 2.4.4 Path loss models 200 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 31 3.3.1 Metrics and procedure 37 4.3 Simulation results and evaluation 37 4.3		1.5	Problem statement	9
2.1 Graph theory 12 2.2 Animal social network data analysis 13 2.2.1 Animal social network metrics 13 2.2.1 Animal social network metrics 14 2.3 Practical examples using animal social network metrics 15 2.3.1 Other evaluation metrics 17 2.4 Wireless technologies 18 2.4.1 Bluetooth Low Energy Overview 18 2.4.2 Long Range (LoRa) Overview 19 2.4.3 Path loss 19 2.4.4 Path loss models 20 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 31 3.3.1 Metrics and evaluation 37 4.3 Simulation results and evaluation 37	2	Bac	kground 1	2
2.2 Animal social network data analysis 13 2.2.1 Animal social network metrics 14 2.3 Practical examples using animal social network metrics 15 2.3.1 Other evaluation metrics 17 2.4 Wireless technologies 18 2.4.1 Bluetooth Low Energy Overview 18 2.4.2 Long Range (LoRa) Overview 19 2.4.3 Path loss 19 2.4.4 Path loss models 20 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 31 3.3.1 Metrics and evaluation 37 4.3 Simulation datasets 37 4.3 Simulation datasets 37 4.3 Isomorphishm 37 4.3.3 B		2.1	Graph theory	2
2.2.1 Animal social network metrics 14 2.3 Practical examples using animal social network metrics 15 2.3.1 Other evaluation metrics 17 2.4 Wireless technologies 18 2.4.1 Bluetooth Low Energy Overview 18 2.4.2 Long Range (LoRa) Overview 19 2.4.3 Path loss 19 2.4.4 Path loss models 20 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 31 3.3.1 Metrics and evaluation 37 4.2 Path loss model setup 37 4.3 Simulation datasets 37 4.2 Path loss model setup 37 4.3 Isolated node detection 38 4.3.3 Betweenness </td <td></td> <td>2.2</td> <td>Animal social network data analysis</td> <td>3</td>		2.2	Animal social network data analysis	3
2.3 Practical examples using animal social network metrics 15 2.3.1 Other evaluation metrics 17 2.4 Wireless technologies 18 2.4.1 Bluetooth Low Energy Overview 18 2.4.2 Long Range (LoRa) Overview 19 2.4.3 Path loss 19 2.4.4 Path loss models 20 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 31 3.3.1 Metrics and evaluation 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3 Simulation results and evaluation 37 4.3 Isomorphishm 37 4.3 Betweenness 39 4.3.4 Node de			2.2.1 Animal social network metrics	4
2.3.1 Other evaluation metrics 17 2.4 Wireless technologies 18 2.4.1 Bluetooth Low Energy Overview 18 2.4.2 Long Range (LoRa) Overview 19 2.4.3 Path loss 19 2.4.4 Path loss models 20 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 31 4 Simulation setup, results and evaluation 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3 Isomorphishm 37 4.3 Isomorphishm 37 4.3.3 Betweenness 39 4.3.4 Node degree 40		2.3	Practical examples using animal social network metrics	5
2.4 Wireless technologies 18 2.4.1 Bluetooth Low Energy Overview 18 2.4.2 Long Range (LoRa) Overview 19 2.4.3 Path loss 19 2.4.4 Path loss 19 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 31 4 Simulation datasets 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Isomorphishm 37 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3			2.3.1 Other evaluation metrics	7
2.4.1 Bluetooth Low Energy Overview 18 2.4.2 Long Range (LoRa) Overview 19 2.4.3 Path loss 19 2.4.4 Path loss models 20 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.1.1 Used simulation datasets 31 3.3.1 Metrics and procedure 31 3.3.1 Metrics and evaluation 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44		2.4	Wireless technologies	8
2.4.2 Long Range (LoRa) Overview 19 2.4.3 Path loss 19 2.4.4 Path loss models 20 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.1.1 Used simulation at a gorithms 31 3.3.1 Metrics and procedure 31 3.3.1 Metrics and evaluation 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 </td <td></td> <td></td> <td>2.4.1 Bluetooth Low Energy Overview</td> <td>8</td>			2.4.1 Bluetooth Low Energy Overview	8
2.4.3 Path loss 19 2.4.4 Path loss models 20 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.1 Metrics and procedure 31 3.1 Metrics and procedure 31 3.1 Ised simulation datasets 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45			2.4.2 Long Range (LoRa) Overview	9
2.4.4 Path loss models 20 2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 20 3.3 Evaluation method and algorithms 30 3.3.1 Metrics and procedure 31 3.3.1 Metrics and procedure 31 4 Simulation setup, results and evaluation 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45			2.4.3 Path loss	9
2.4.5 Log-normal shadowing 22 2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.1.1 Metrics and procedure 31 3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 31 3.3.1 Metrics and procedure 31 4 Simulation setup, results and evaluation 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47 </td <td></td> <td></td> <td>2.4.4 Path loss models</td> <td>0</td>			2.4.4 Path loss models	0
2.5 Summary 23 3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 31 3.3.1 Metrics and procedure 31 4 Simulation setup, results and evaluation 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47			2.4.5 Log-normal shadowing	2
3 Methodology 26 3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.1 Metrics and procedure 31 3.1 Used simulation setup, results and evaluation 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47		2.5	Summary	3
3.1 Simulation method 26 3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 31 3.3.1 Metrics and procedure 31 4 Simulation setup, results and evaluation 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47	3	Met	hodology 2	6
3.1.1 Used path loss models 28 3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 31 3.4 Simulation setup, results and evaluation 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3 Simulation results and evaluation 37 4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47		3.1	Simulation method	6
3.2 Experimental design 30 3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 31 3.3.1 Metrics and procedure 31 4 Simulation setup, results and evaluation 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47 <td></td> <td></td> <td>3.1.1 Used path loss models</td> <td>8</td>			3.1.1 Used path loss models	8
3.3 Evaluation method and algorithms 31 3.3.1 Metrics and procedure 31 4 Simulation setup, results and evaluation 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47		3.2	Experimental design	0
3.3.1 Metrics and procedure 31 4 Simulation setup, results and evaluation 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47		3.3	Evaluation method and algorithms	1
4 Simulation setup, results and evaluation 37 4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47			3.3.1 Metrics and procedure	1
4.1 Used simulation datasets 37 4.2 Path loss model setup 37 4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47	4	Sim	ulation setup. results and evaluation 3	7
4.2Path loss model setup374.3Simulation results and evaluation374.3.1Isomorphishm374.3.2Isolated node detection384.3.3Betweenness394.3.4Node degree404.3.5Density444.3.6Component detection454.3.7Communities47		4.1	Used simulation datasets	7
4.3 Simulation results and evaluation 37 4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47		4.2	Path loss model setup	7
4.3.1 Isomorphishm 37 4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47		4.3	Simulation results and evaluation	7
4.3.2 Isolated node detection 38 4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47			4.3.1 Isomorphishm	7
4.3.3 Betweenness 39 4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47			4.3.2 Isolated node detection	8
4.3.4 Node degree 40 4.3.5 Density 44 4.3.6 Component detection 45 4.3.7 Communities 47			4.3.3 Betweenness	9
4.3.5 Density			4.3.4 Node degree	0
4.3.6 Component detection 45 4.3.7 Communities 47			4.3.5 Density	4
4.3.7 Communities			4.3.6 Component detection	5
			4.3.7 Communities	7

5	Experimental setup, results and evaluation 5				
	5.1	.1 Hardware and software setup			
	5.2	Data g	athering approach	54	
	5.3	Experi	mental data gathering	55	
	5.4	Experi	mental test results and evaluation	57	
		5.4.1	Isomorphishm	61	
		5.4.2	Isolated node detection	61	
		5.4.3	Betweenness	61	
		5.4.4	Node degree	64	
		5.4.5	Density	64	
		5.4.6	Component detection	67	
		5.4.7	Communities	67	
6	Conclusion 6			69	
	6.1	Conclu	usion	69	
	6.2	Limita	tions and future work	71	
	6.3	Feedb	ack from a biologist	73	
Α	Adjacency matrices degree evaluation 7				
В	Adjacency matrices component detection 8				
С	Adjacency matrices community evaluation 8				

List of Figures

1	Example of a typical graph network, showing vertices (V) (red dots) and edges	
	(E) (blue links)	12
2	Example of graph network isomorphism	13
3	Example of betweenness in a graph network	14
6	Path loss and wireless coverage [1]	23
4	Examples of social network analysis usage in wildlife conservation (image	
	taken from <i>Snijders et al.</i> [2])	24
5	Mutual information where $I(X : Y)$ represents the mutual information and	
	H(X Y) and $H(Y X)$ the non-mutual information	25
7	Overview of the system simulation tool	26
8	Photos of the BLE test location	28
9	Overview of the test location, a and b correspond to the place the photos from	
	Figure 8 were taken. Image taken from Google Maps	28
10	Comparison of the simplified path loss model, the Okumura-Hata model and	
	the Hata-COST model	29
11	Photos of the LoRa test area	30
12	Overview of the evaluation design	31

13	Simulation results, isolated nodes: Offset of the number of isolated nodes in	
	the actual graph compared to the estimated graph	39
14	Simulation results, isolated nodes: Same node score	40
15	Simulation results, betweenness: Offset of the number of nodes with high	
	betweenness in the actual graph compared to the estimated graph	41
17	Simulation results, degree: Offset of the number of nodes with high degree in	
	the actual graph compared to the estimated graph	41
16	Simulation results, betweenness: Same node score	42
18	Simulation results, degree: Same node score	42
19	Graph of the actual network (a) and the graph of the estimated network (b) for	
	one timeframe	43
20	Simulation results, density: Difference between the actual density and the	
	estimated density	45
21	Simulation results, component detection: Offset of the number of components	
	in the actual graph compared to the estimated graph	46
22	Graph in the graph based on the actual distances (a) and in the graph based	
	on estimated distances (b) for one timeframe	47
23	Simulation results, community detection: NMI values	48
24	Communities found in the graph based on the actual network (a) and in the	
	graph based on the estimated network (b) for one timeframe	49
25	Communities found in the graph based on the actual network (a) and in the	
	graph based on the estimated network (b) for one timeframe	49
26	The Raspberry Pi model 3B combined with the LoRa/GPS hat	52
27	File with BLE scan results. The first number represents the minute of the hour,	
	the second number represents the detected node id and the third number is	
	the signal strength	54
28	Scenario's 1 – 4	55
29	Scenario's 5 – 8	56
30	The actual and estimated network for scenario 1	58
31	The actual and estimated network for scenario 2	59
32	The actual and estimated network for scenario 3	59
33	The actual and estimated network for scenario 4	59
34	The actual and estimated network for scenario 5	60
35	The actual and estimated network for scenario 6	60
36	The actual and estimated network for scenario 7	60
37	The actual and estimated network for scenario 8	61
38	Experimental results, isolated nodes: Offset of the number of isolated nodes	
	in the actual graph compared to the estimated graph	62
39	Experimental results, isolated nodes: Same node score	63
40	Experimental results, betweenness: Offset of the number of nodes with high	
	betweenness in the actual graph compared to the estimated graph	63
41	Experimental results, betweenness: Same node score	64

42	Experimental results, degree: Offset of the number of nodes with high degree		
	in the actual graph compared to the estimated graph	65	
43	Experimental results, degree: Same node score	66	
44	Experimental results, density: Difference between the actual density and the		
	estimated density	66	
45	Experimental results, component detection: Offset of the number of compo-		
	nents in the actual graph compared to the estimated graph	67	

List of Tables

1	Methods of animal tracking	10
2	Adjacency matrix corresponding to the graph in Figure 1	13
3	Social Network Indicators	16
4	LoRa spreading factors with bit rate and transmission (TX) power	20
5	Variables for simplified path loss models	21
6	Path loss exponents for different environments	21
7	Variables for Okumura-Hata model	21
8	Example of the actual distance matrix with four nodes	26
9	Example of the estimated distance matrix with four nodes	27
10	Example of an adjacency matrix based on the actual distance matrix (Table	
	8) for 4 node pairs	27
11	Example of an adjacency matrix based on the estimated distance matrix (Ta-	
	ble 9) for 4 node pairs	27
12	Parameter values used in the simulation for BLE	37
13	Parameter values used in the simulation for LoRa	38
14	Node degree comparison between the actual and estimated network corre-	
	sponding to Figure 19	44
15	Component comparison between the actual and estimated graph	47
16	Community comparison of actual and estimated communities with NMI of 1 .	50
17	Community comparison of actual and estimated communities with NMI of 0.95	51
18	Performance of BLE and LoRa for different metrics	72
19	Adjacency matrix for the actual network corresponding to Figure 19a	79
20	Adjacency matrix for the estimated network corresponding to Figure 19b	80
21	Adjacency matrix corresponding to the graph in Figure 22a	81
22	Adjacency matrix corresponding to the graph in Figure 22b	82
23	Adjacency matrix for the actual communities with NMI = 1	83
24	Adjacency matrix for the estimated communities with NMI = 1	84
25	Adjacency matrix for the actual communities with NMI = 0.95	85
26	Adjacency matrix for the estimated communities with NMI = 0.95	86

1 Introduction

Social network analysis is a topic that most people will have heard about, mostly due to online social network platforms like Facebook, LinkedIn, Instagram and Twitter. Those networks are very popular and used by people to, for instance, share content (such as photos and videos), keeping in contact with friends and family and participate in discussions. When the popularity of the smartphone increased (approximately around the year 2007) [3], it became possible for users to not only visit the social network using their computer at home but also when users are outside, making it easier to share content and keep in touch with others. Where in the past users had to use their (home) computer for participating in the network and had to transfer photos/videos from their digital camera to their computer to be able to share it, users were now able to participate and share content directly from their smartphone. Facebook (founded in 2004) had 58 million monthly returning users in 2007 (in this year the first smartphones with a large touchscreen were released, the iPhone and LG Prada) [3], this became 145 million monthly returning users in 2008 (the year when the first smartphone with Android was released) [3], 360 million monthly returning users in 2009 [3], all the way up to 1.23 billion monthly returning users in 2013 [3] and 2.2 billion monthly returning users in 2018 [4].

Networks like Facebook use social network analysis to suggest possible friends to users, to suggest events where a user might be interested in and for directed advertising, among others. Possible friendships can be suggested by taking into account with whom the user is befriended (by looking at the (amount of) mutual friends between two users), the location of a person and how much time the person spends at specific locations. For instance, when a group of people arrive at a bar around the same time on a regular basis and/or if they work at the same company, they probably know (about) each other. Event suggestion and targeted advertisement can be based on for example interests or the location (are events taking place nearby or at a place where the user has been before) [5].

Social network analysis is not only used for the analysis of social contact between humans, but also in applications involving animals. *Animal social network analysis* (animal SNA) can help biologists understand social and ecological interactions between animals. Animal social network studies can be placed in four categories. Those are research to (1) *the social structures in a group* [6], to study (2) *the causes and consequences of the different behaviour of individuals in a group* [6], studies to (3) *social processes in a group, for information or disease spreading* [6] and to study (4) *the relationship between the environment and the network structure* [6]. Conservation biologists can use animal SNA to test hypothesises, for evaluation purposes or to gather information about the population status. The benefit of using animal SNA in conservation applications is that SNA can provide an understanding of the overall social structure and can be used to detect important changes in the social network [2].

Only twenty years ago, over 10.000 animal species were threatened with extinction,

where nowadays this number is above 25.000 [7]. This can partly be explained because more animals are being monitored, but it is mostly caused by humans. From all the mammals, one in five species is threatened with extinction, not only due to poaching but other causes can be deforestation, habitat changes or decreasing territory size [7], [8].

In wildlife conservation, animal social network analysis can be used to quantify social structures in a group of animals, helping to predict how populations will respond to certain environmental disturbances that could cause a population to fragment or crash. When monitoring a population of animals using social network analysis, changes in the network can be used for (early) identification of warning signs of population fragmentation or population collapse. When, for example, the connectivity between individuals changes drastically, a cause can be the fragmentation of habitat, which influences the encounter rates and likely results in social interaction changes, mate choice options and anti-predator behaviour. All of this can influence the fitness of individuals [2].

When animals live close together in the last (small) piece of suitable habitat, this possibly results in more aggressive encounters, leading to higher stress levels and higher injury rates, making it easier for contagious diseases to spread. Monitoring and adjusting the social structure can help in finding solutions for those problems and evaluating them. Examples of applications are identifying which animals should be strategically vaccinated to stop the fast spreading of diseases or to see what happens when a group of animals is relocated to a new location, by monitoring what happens to the social structure [2].

While there are a number of options for animal monitoring (discussed in chapter 1.4), this project assumes that biotelemetry tags (electronic tags capable of radio communication, further discussed in section 1.4) are used for the monitoring of animals. Biologists aim to use tags smaller than 5% of the body weight of the animal [9], in order to minimise the effect on the behaviour and the survival chances of the animal. In the last decade, the weight of tags equipped with wireless communication has dropped from 250 grams to 20 grams and the resolution of those tags increases by approximately one order of magnitude every five years, making it possible to tag smaller animal species. Despite this, about 70% of bird species and 65% of mammal species still cannot be tracked using these tags, making it important to keep reducing the tag size [9].

Social interactions between animals can be recorded by using either GPS for localisation, or proximity sensors, which uses ultra-high-frequency (UHF) transmitters and receivers to register when animals are in proximity of each other. Conventional GPS units regularly need to download and configure satellite data before the animal location can be derived. Low-power GPS units can bypass this by post-processing the data on computers when the data is retrieved, resulting in less energy consumption compared to conventional GPS units [10]. This project assumes that the used tags are equipped with Bluetooth Low Energy (BLE) and LoRa for wireless communication. Both technologies are known for their low energy consumption [11].

Section 1.1 discusses the history of human social network analysis, followed by 1.4, which discusses how animal movement data can be gathered and concludes with the problem statement in 1.5.

1.1 Background of human social network analysis

Social network analysis is a concept originating from approximately the early 1900s and notable advances in the field were made in the 1940s and 1950s, with one of them being the use of matrix algebra and graph theory (further discussed in 2.1) to formalise fundamental concepts such as groups and social circles in network terms, making it possible to objectively discover groups in network data [12]. Other notable advancements were made by studies to social network structures, resulting in more knowledge about network centrality, eventually leading to a solution to the small world problem (if two persons are randomly selected, what are the chances that they know about each other? And what is the minimum number of people needed to link them together?). This led to the six degrees of separation theory, stating that the minimum number of people needed to link two persons together is five [12].

A notable advancement from the 1970s is the theory about the *strength of weak ties* (also known as the *strong triadic closure principle*) by Granovetter, who described that there are two types of *social ties*, strong ties (e.g. friends) and weak ties (e.g. acquaintances) [13]. When he was working on his PhD thesis, Granovetter had the idea to interview people who had just changed from their job and employer and asked them how they learned that the job was available. It turned out that most people heard about the vacancy trough acquaintances instead of close friends [13], [14].

Granovetter argued that, if two people in a network have a friend in common, there is an increased likelihood that they will become friends in the future. He further argued that it is impossible for two individuals, having a strong social tie with a common friend, to not know each other. This leads to the situation that a person with two close contacts receives information in a redundant way (what person A tells him/her, is likely the same as the information coming from person B), while weak ties are more likely to be a source of new information, acting like a bridge between the persons own network and that of the acquaintance. Thus, this is the reason why people are likelier to hear information about vacancies trough acquaintances instead of close friends [13], [14]. *Hillinan* did research to the social ties between schoolchildren. It turned out that ties (friendships) are more likely to drop when they are cross-sex or cross-race, and ties are stronger if they are demographically similar. More research has been done by *Popielarz and McPherson* in 1995, who showed that the more a group member differs from the rest of the group, the weaker the tie and the more likely it is that the member will leave the group [15].

A theory related to social ties is the *homophily theory* [15], stating that people have a higher chance of getting in contact with each other when they have similar interests. It also states that the social tie between persons becomes stronger when two persons share

multiple interests or relationship types (e.g. sporting together, being class mates and being friends at the same time). Two types of homophily are distinguished, *status homophily*, containing ascribed characteristics including characteristics like race, ethnicity, sex and age, and *value homophily*, containing acquired characteristics like religion, education, occupation and behavioural patterns. The most basic reason for homophily is the geographic location. People are more likely to have contact with people who are geographically closer than those who are living further away. The reason for this is effort. *Zipf* stated in 1949 that it takes more energy to contact people who are far away than with people in the immediate geographic environment. While technologies like e-mail and telephone have made it easier to keep in contact with others, and thus have lowered the effort, *Verbrugge* found in 1983 that residential proximity is still the best predictor of how often friends come together. People seeing each other face-to-face on a regular basis are still likely to have a stronger relationship than people having regular contact trough e-mail or the phone [15].

The homophily theory is used in the analysis of *contact patterns*, which in turn is used to analyse (and predict) the social contact between different people. Contact patterns includes two perspectives, the centrality and community perspective. Centrality indicates that a person is important in the network, having links to a lot of other people and frequently meets with them. Communities state that people are organised into groups, according to their social relations, where different people can be part of different communities (a person can be part of the co-worker community in the daytime, while the person is part of the sports club or family community in the evening) [15].

In the past decade, *online social networking* became a popular way for people to keep in contact, sharing photos and videos or participate in discussions, with Facebook (2.2 billion monthly returning users in 2018 [4]), YouTube (1.5 billion monthly returning users in 2017 [16]), Instagram (1 billion monthly returning users in 2018 [4]) and Twitter (328 million monthly returning users in 2017 [16]) as well-known examples of social networks. Those networks are not only accessible trough a website, but can also be used on smartphones, making it possible for users to be connected with the network whenever they want, leading to increase of social network usage. Online social networks are based on user profiles containing attributes like, for example, the user's geographical location, attended schools, (past) jobs, interests or messages to and from other persons. While not every user provides all those attributes to the network, if only 20% of the users provides those attributes, it is possible to infer the attributes of the remaining users with an 80% accuracy [17].

User's profiles, including their attributes and activities, provide a valuable insight into a user's behaviour, experiences, opinions and interests, giving information about the user's personality, behaviour and mental processes, leading to enormous opportunities for (new) social network analysis applications. Online social network websites, and especially Facebook, have been criticised over the past few years, mainly due to privacy concerns [18], with the Cambridge Analytica scandal being the most recent scandal. Cambridge Analytica described itself as a "political analysis firm", claiming to build psychological profiles of voters

and using those profiles to help their clients win elections [19]. The firm was able to gather detailed personal information of, according to Facebook, 87 million Facebook users [20] and is accused of using this information to influence The US presidential elections of 2016 [21].

While online social networks offer a lot of future possibilities in the field of social network analysis, given the enormous amount of user data, it also raises a lot of ethical questions. Apart from the possible election influence, another example of a recent ethical question is the *"Predicting Life Changes of Members of a Social Networking System"* patent filed by Facebook, where social network analysis is used to predict "life-changing" events in a persons life like a marriage, a new job, birth of a child or a person's death [22]. While social network analysis is an old concept, thanks to recent technological advancements like online social networks and the broad adoption of the smartphone make social network analysis more relevant than ever. Social network analysis has nowadays been adopted in a wide variety of fields, including psychology and economics [12].

1.2 Indicators of human social interaction

To build a social network graph, indicators of social interaction are needed to find out which persons have contact with each other. The main indicators of social contact between humans are *social contact*, *human mobility* and *proximity* [23].

Social contact includes for example file transfer, e-mails and phone calls between users or information available on online social networks, e.g. wall posts or private messages between users [23].

Human mobility is used for social contact prediction. In potential, the movement and mobility of people is highly predictable, since most people have a routine of travelling to work, to home or places like the supermarket at a certain schedule. Mobility (or movement) information can be used to determine to which social communities a specific person belongs, by distinguishing the preferred locations of a person. Mobility information can also be used for creating social graphs, to discover the communities relevant to a person or for modelling an agenda. This agenda can be used to predict movement patterns based on repeating (daily or weekly) activities, indicating which people a person might meet or has a high possibility of meeting at certain places and moments [5].

For mobility, three properties have to be taken into account, the *spatial properties, temporal properties* and *connectivity properties* [5].

The **spatial properties** define the distance that a person travels to locations. Most people usually travel in close vicinity to their homes, only a few people make long journeys on a frequent basis. **Temporal properties** takes into account how much time a person spends at a location and how often the person returns. At last, the **connectivity properties** define how often a person meets with other people at specific locations and the contact duration [5].

The *proximity property* infers social contact between people, by measuring the distance between them. Individuals having a conversation are approximately 0.5 to 2.5 meters apart from each other [24].

1.3 Applications of social network analysis

This section discusses a couple of applications using human social network analysis that use (some of) the aforementioned concepts and theories. In the 1940s, researchers interviewed 1050 adults, living in 50 different Northern Californian communities with varying degrees of urbanism, about their social relations. Respondents had to identify people with whom they had some kind of relationship and tell about them. The researchers found out that dense urbanism did reduce network density, which was known to be negatively related to the psychological health, satisfaction level and overall well-being [25].

Where in the past researchers had to interview people about their social relations, technological advances like the smartphone and online social networks make it possible to electronically gather data for the purpose of social network analysis. Using social contact indicators, *Kahanda and Neville* use friendship information from Facebook in combination with three different kinds of *transactional data*, i.e. public posts on prole pages, pictures and group membership, to define the strength of the social ties between people. Their experimental results indicate that transactional events (*e.g. communication, file transfer*) are useful for the prediction of tie strength and that its important to consider the transactional events in context of the user behaviour [23].

Stehl et al. [26] use contact patterns, focused on the pattern constraints and temporal aspects (i.e. how much time people spend on certain locations) to gather contact information, and use this information in a simulation to gather information about the spread of infectious diseases in a population. The authors used RFID tags, voluntarely worn by 405 out of 1200 conference attendees at a 2-day conference with a 12 hour measurement on day 1 and an 8 hour measurement on day 2, to determine how often and for how long people have contact with each other and use this information to simulate the spreading of diseases. The authors compared the data collected on day 1 of the conference with the collected data on day 2 and found out that the statistical distributions of the number and duration of contacts and the link weights were similar. Limitations of the experiment, according to the authors, are that individuals are not followed outside the zone with RFID readers, which influences the number of contacts and thus influences their findings. Furthermore, the volunteers were followed for a limited amount of time, and not during a 24-hour time period. The limited amount of volunteers (34%) is also a limitation. Nevertheless, the authors prove that data collected with RFID tags is an effective way to simulate the spreading of diseases [26].

Isella et al. [27] analyse contact patterns, gathered by using badges with embedded RFID chips to gather data about human behaviour. They did this at a museum setting, where 100% of the visitors participated because participation was mandatory, and at a conference

setting, where about 75% of the visitors volunteerd in participating. The authors state that there is a clear difference in the behaviour of humans in the two settings. Museum visitors spend a limited amount of time at the location and follow a rather pre-defined path. Visitors of a conference spend all their time at the conference location and freely move around between areas. Despite the differences in the settings, they show that the distribution of the contact event duration as well as the face-to-face interactions by two individuals are very similar. According to the authors, future work can be to collect data in a conference setting when all conference visitors are participating, since this might have influenced their findings [27].

Gnois et al. [28] also use contact patterns to reveal the spreading of infectious diseases in an office setting, where about 66% of the employees participated, with the goal to find a low-cost vaccination strategy. They gather their data in the same way as *leslla et al.*, and also compare their data to the data of *lsella et al.*. They show that the distributions of the data are very similar, even though the settings are very different. The authors aim to identify so-called "linkers", people who act like a "bridge" between different (social) groups and play an important role in disease spreading. They were able to identify three different behaviours, residents (mostly internal contact), wanderers (mostly external contact) and linkers (contact with both internal and external nodes) by building a global contact network (with weighted links, based on contacts, between nodes), based on contact data over two weeks, and by calculating the contact time between individuals. The authors state that, while their research shows that they can identify linkers, it is also possible to identify them by looking at the human behaviour, activities or attributes in the organizational chart, which is a lot easier. They also state that their research should be repeated in other communities like schools or hospitals [28].

Yu et al. [29] use GPS trajectory data to predict the future locations of persons on a campus. Their experiments showed a 90% predicting accuracy with a five-minute prediction error for the arrival time. Using this information, the authors proposed two mobile phone applications. The first one is *HelpBuy*, an application where a user requests others to buy what he/she needs and deliver it to him/her, based on their future location predictions. The second one is *EaTogether*, an application that predicts when friends can meet at restaurants (by predicting at what time all the friends are near the restaurant) and encourages them to have a meal together, with the goal to encourage face-to-face meetings. The collected data was collected in a very specific scenario (a campus) and the authors aim to try the same method on other situations and other datasets [29].

Cheng et al. [30] use the information transmitted by devices over WiFi. The authors make use of the fact that modern operating systems keep a Preferred Network List (PNL), which contains the names of previously accessed networks. The mobile devices regularly broadcast the SSID's in the PNL in plain text, mostly in the order with the most or the last probed networks first. The authors combined this with the locations of each WiFi network in each campus building. This allowed them to find out where devices and their owners have been and map this to locations. They combined this information with mobility traces,

which are used to find out if and when users are connected with an access point. In order to differentiate public WiFi AP's (e.g. Eduroam) from a home network (e.g. wifiCoen), different weights are are assigned to different SSIDs, because the relationship can be weaker when two users share a public WiFi AP. The authors were able to infer 30 relationships in a time-span of 30 days. Shortcomings are that the broadcasted PNL, where the authors rely on, is mostly partially broadcasted, because the client stops broadcasting the PNL when a known network has been found [30].

1.4 Gathering animal movement data

While interest in human SNA has seen an enormous growth since the 2000s (the amount of papers discussing the topic has almost tripled between the period between 2000 — 2009 [12]), biologists are nowadays also adopting SNA in fields like movement ecology [31] and wildlife conservation [2], by analysing *animal movement data*. The following section discusses how the data is gathered and how this data is used for animal social network analysis.

Animal movement data can be gathered by different methods, using passive or active monitoring methods [31]. There are two basic ways of animal monitoring, the Lagrangian approach where a specific individual is monitored and records the location of the animal, for example with a GPS tag, and the Eulerian approach, where a specific location is monitored and the movement of all the animals passing through the location are recorded [32]. Used methods for data gathering are placing *camera traps* [32], acoustic fixed arrays [33], [34], mark-recapture [35], biologging [9], [36] and biotelemetry [9], [36] [31].

For the *camera trap* method, motion sensitive cameras are placed in the area of interest, offering a non-invasive way of monitoring since most animals wont know that a photo is taken. Camera traps can function for weeks without needed attention, so data gathering requires low labour. The photos taken by camera traps can not only be used to capture the presence (or absence) of animals, but also captures their behaviour. Camera traps can also be used to determine the local animal density, which becomes more valuable when data is gathered over years or across different sites. Downsides to camera traps are the limited live transmission of data (live data transmission requires a much larger battery or a solar panel) and the fact that it only captures animals in front of the camera, so it is likely to miss animals [32].

Using *acoustic fixed arrays* (or acoustic monitoring) is a non-invasive way of animal monitoring, using microphones placed in the environment to study animals, allowing biologists to estimate the positions of animals. This provides spatial context for monitoring and measuring animal movement. Multiple animals can be studied simultaneously while human observers are absent from the area. Acoustic monitoring is suitable for the monitoring over long time periods and can be used for monitoring over night or in thick vegetation, where visual tracking is difficult or impossible [33], [34].

Acoustic monitoring cannot be used for silent animals and because sound attenuates rapidly, it requires that microphones are positioned around and close to the target animals to collect suitable recordings. Spatial acoustic monitoring also requires precise coordination of the recordings from each microphone, requiring that the clock of the microphones must regularly be synchronised on a millisecond level. Some researchers found a solution to this problem by using kilometres of cable to connect microphones to a central recorder, increasing the amount of labour to set this up [33], [34].

In the *mark-recapture* method, animals are caught and marked with for example metal bands, colour bands, ear tags or toe clips and released, making it possible to identify the animals at a later point in time when they are re-captured or re-sighted. The method makes it possible to gather information about characteristics of individuals (such as age or sex), population changes over time and the impact of management actions. Disadvantages are that tags can get lost, animals can disappear or can be hard to re-catch. There is no information of the animal behaviour between sightings [35].

Biologging and **biotelemetry** are methods that can provide detailed information, including behaviour and the distribution of animals in space and time. While animals in both biologging and biotelemetry are tagged with an electronic device gathering information, the fundamental difference is that in biologging, the data is retrieved when the tag is retrieved, while in biotelemetry data is transmitted between the tag and a receiver by using a radiolink. Both methods can be used for the monitoring of biological and environmental variables and make it possible to study free-living animals in their natural environment. This is particularly relevant for endangered species, where biologists dont want to remove animals from the environment. The used tags can be equipped with sensors for e.g. the monitoring of the heart rate, temperature, proximity sensors, accelerometers or cameras, for detailed continuous animal monitoring without human support. This eliminates data gaps and enables the monitoring of animals living in large habitats, animals occupying habitats that are hard to reach or rapidly moving animals [9], [36].

Limitations of biologging are that tags must be retrieved in order to collect the data, while biotelemetry is able to transmit data on a regular basis, with the downside that the tags need a bigger battery to compensate for the energy consumption of data transmission or have a shorter lifetime [36]. For the tags used in this kind of research, biologists aim to use tags smaller than 5% of the body weight of the animal, in order to minimise the effect on the behaviour and the survival chances of the animal [9]. An overview of data gathering methods can be seen in table 1.

1.5 Problem statement

The motivation for this research is to find out whether it is possible to use proximity, inferred from the wireless radio links signal level, and combine this with animal social network analysis to infer information about the social network of animals. If it turns out that it is possible

Technique	Methods	Advantages	Disadvantages	Reference
Camera trap	Cameras are placed	Noninvasive,	Limited live	[32]
	on strategic locations	easy to deploy,	transmission of	
	in order to capture an-	Also records an-	data (lack of	
	imal activity	imal behaviour	battery power),	
			only captures	
			what is in front	
Accustic	Mierophonoo	Nen invesive	of the camera	[00] [04]
ACOUSTIC	Microphones are	NON-Invasive,	for eilept opi	[33] [34]
inxed array	placed on strategic	suitable for mon-	nor silent ani-	
	places to identify		mais, spalla	
	species by sound	areas, ambient	hord it is hord	
		ho takon into ao	ta actimata tha	
			no estimate the	
		for moscuring	sound source	
		biodiversity	beavily de-	
		biodiversity	neads on clock	
			synchronisation	
Mark and re-	Animals are (in most	Recapturing	Animals need	[35]
capture	cases) captured.	not alwavs	to be marked	[]
	marked and released.	necessary (re-	(in most cases),	
	They are later recap-	sighting), gives	marks can get	
	tured and checked	information	lost, animals can	
		about population	disappear or can	
		changes over	be hard to catch	
		time		
Bio-logging	Data collecting tags,	Long battery	Invasive, tags	[36]
	attached to an animal.	time (no data	can get lost, no	
	Tag is released from	transmission), a	real-time data	
	the animal when the	lot of measure-	transmission	
	battery is empty. Data	ment options		
	is collected when the	(e.g. location,		
	device is retrieved	speed, heart-		
Riotelemetry	biologging devices	Deat)	Invasive tage	[36]
Dioteienieu y	combined with	surement on	can get loet loes	[50]
	telemetry to cond	tions (a a	hattery time than	
	data	location speed	bio-logging	
	Juiu	heartbeat)	210 1099119	
		real-time data		
		transmission		
		possible		

Table 1: Methods of animal tracking

to use LoRa and/or BLE for network analysis, it eliminates the need to have an additional proximity sensor or GPS chip for gathering social network data. Being able to omit these, means that less components need to be powered and less size is needed for the components, which can be used to make the tags smaller, add additional sensors or increase the battery size. LoRa consumes 1.4mA when in standby, 10.5 mA when in receiving mode and 18 mA when in transmit mode [37]. BLE consumes 10.1 mA when in receive mode and 10.8 mA when in transmit mode [38]. GPS consumes 25 mA when determining the location [39]. The values for GPS and LoRa are taken from the datasheets corresponding to the hardware that is used for test purposes. The hardware is discussed in section 5.1. The values for BLE are taken from the datasheet corresponding to Nordic NRF52 devices, which are used for the validation of the BLE transmission range.

The challenge in this is that the signal strength between two wireless radios is not a reliable source for distance estimation. Due to phenomena like path loss, reflection and scattering interfere with the transmitted signal. Therefore, the received signal strength is not a precise rendition of the true distance between the radios, but rather a rough estimation.

The following research question has been defined, with corresponding subquestions to help answer the research question:

Can LoRa and/or BLE replace GPS and proximity sensors in animal tags for inferring social interaction between animals?

- · What is social network analysis and its aspects?
- What is animal social network analysis and how is it performed?
- What is the performance of the techniques (LoRa or BLE) in animal social network analysis?

Those questions are answered by using a simulation tool, simulating the radio signal between animals. An hardware implementation is used for validation.

This document is structured as follows: chapter 2 discusses the history of social network analysis and recent developments in the field, an explanation of graph theory and animal monitoring, discusses animal social network analysis and gives information about Bluetooth Low Energy and LoRa. Chapter 3 explains how the simulator works, the used path loss models and discusses the hardware and its related tests. In chapters 4 and 5 the results are discussed and evaluated. The thesis ends with a conclusion and recommendations in chapter 6.

2 Background

This chapter aims to cover all the relevant background information needed to understand the performed research. It covers the history and background of social network analysis and recent developments in the field, explains how social network analysis is used for inferring social contact between humans, the basics of graph theory, an explanation of animal monitoring and animal social network analysis. It concludes with an explanation of Bluetooth Low Energy, LoRa, path loss and shadowing.

2.1 Graph theory

As noted in section 1.1, an important aspect of social network theory is graph theory. This section discusses the basics of graph theory and adjacency matrices.



Figure 1: Example of a typical graph network, showing vertices (V) (red dots) and edges (E) (blue links)

Graph theory is based on graphs, consisting of vertices (also called nodes) (V) and edges (E) connecting those vertices. A graph G contains the sets (V,E) with the vertices labelled as $V = v_1,...,v_n$, and the pair of edges labelled as $E = (v_1,v_n)$, often also labelled as $E = e_1,...,e_m$. If V and E are empty, the graph is a null graph and two vertices in a graph are adjacent if they are connected by an edge [40] [41]. Figure 1 shows an example of a graph with its vertices and edges.

Another way to represent a graph is by using an adjacency matrix. For the matrix, the assumption is made that the vertices are number 1, 2, ..., |V| in an arbitrary manner. The adjacency matrix representing a graph G = (V,E), consists of a |V|X|V| matrix $A = (a_{ij})$, where $a_{ij} = 1$ if $(i, j) \in E$, otherwise $a_{ij} = 0$ [40]. For a weighted graph, the weight of the graph can be stored instead of a 1. As an example, the adjacency matrix of the graph in Figure 1 is shown in Table 2.

Isomorphism

Isomorphism is used to check if two graphs are exactly the same. The graphs G and G' are isomorph if the vertices of G can be relabelled to to be vertices of G', while maintaining

	1	2	3	4	5	6	7	8	9
1	0	1	1	0	0	0	0	0	0
2	1	0	1	1	1	0	0	0	0
3	1	1	0	0	1	0	0	0	0
4	0	1	0	0	1	0	0	0	0
5	0	1	1	1	0	0	0	0	0
6	0	0	0	0	1	0	1	1	1
7	0	0	0	0	0	1	0	0	1
8	0	0	0	0	0	1	0	0	1
9	0	0	0	0	0	1	1	1	0

Table 2: Adjacency matrix corresponding to the graph in Figure 1

the same edges in G and G' [40]. Figure 2 shows an example of isomorphism. Graph G (left) is isomorph with graph G' (right). Even though the graphs are different, both graphs have two nodes with degree two and two nodes with a degree of three.



Figure 2: Example of graph network isomorphism

2.2 Animal social network data analysis

Using the data gathered by the aforementioned methods, *animal social network analysis* can be performed, giving insight to the social structure of a population. This social structure can influence how populations respond to environmental changes (what happens when certain animals are removed from the group? what if the habitat size decreases?), used for understanding the population and can potentially be used for manipulating the dynamics of the population, with the goal to increase their survival chances [2]. An important aspect regarding this is habitat connectivity, covering the fragmentation and distribution of the habitat and can be a critical component in the field of conservation. A problem is that there is no exact definition of habitat connectivity, since it can be measured at a patch or landscape scale and can be structurally or functionally defined. Graph theory provides a solution to this problem because it provides a framework for analysis on different scales, can be used in a dynamic

way and offers a framework to quantify connectivity and flow in social networks [42].

Various metrics are used for the analysis of animal social networks. Those are **be***tweenness centrality, density, community detection, component detection* and *node degree* [2], [31]. The following section gives a description of the metrics. Section 2.3 discusses examples of how the metrics are used in animal social network analysis.

2.2.1 Animal social network metrics



Figure 3: Example of betweenness in a graph network

Betweenness is an indication of the importance of a node in a graph, indicating the flowpotential of, for example, information or diseases [2]. It calculates the number of shortest paths between all the possible pairs of nodes in the graph that traverse the node. An example is shown in Figure 3. The edge between nodes A and B has the highest betweenness in this graph because it connects the left community (the group of nodes on the left) with the community on the right (the group of nodes on the right), thus all the links connecting the nodes in the left community with those in the right community pass through the edge between A and B, resulting in a high betweenness value for nodes A and B [41]. Nodes with a high betweenness are likely to connect largely independent communities [6].

The *density* of a graph is defined by the number of edges in a network divided by the total possible edges or, in a weighted graph, the sum of edge weights divided by the number of possible edges [6]. The density calculation for an undirected graph is shown in equation 1, the one for a directed graph in equation 2 with E the number of edges and V the number of vertices in the graph. The maximal density is 1 (when every node is connected) and the minimal density is 0 (no connection at all) [43]. In practice, the graph density is used in the analysis of movement patterns [31], e.g. the connectivity in a school of tuna [44].

$$D = \frac{2E}{V(V-1)} \tag{1}$$

$$D = \frac{E}{V(V-1)} \tag{2}$$

Community detection is used to detect different communities within the network, along with their members, to indicate movement strategies or for the detection of movement corridors [31]. A community is a group of closely connected individuals with a less dense

connection to the rest of the population. Community detection indicates how socially integrated the population is. The more nodes in a community, the more socially integrated the population, but it is more group-oriented when there are more (smaller) communities in the network [2].

The *Girvan–Newman algorithm* is used for the detection of communities in a graph, which is the most popular community detection algorithm. Since the betweenness value is the highest for edges that connect communities, the algorithm selects the edge with the highest betweenness centrality value and removes this edge from the graph. After this, it recalculates the betweenness values in the graph and again removes the edge with the highest betweenness, with the result that different communities become isolated [45]. The algorithm uses betweenness to identify the edges that need to be removed because this method has the best performance [46]. Recalculating the betweenness turned out to be a crucial step. Without betweenness recalculation the performance of the algorithm drops drastically [46].

The algorithm has the following steps [46]:

- 1 Compute betweenness centrality for all edges
- 2 Remove edge with largest betweenness centrality
- 3 Recalculate the betweenness centrality in the graph
- 4 Repeat from step 2

If the algorithm would be used on the graph from Figure 3, the result is that there are two communities, the edge between node A and B will be removed because of the high betweenness.

Component detection is similar to community detection. Where groups found in community detection have a connection to the rest of the population, component detection measures the groups (with at least more than one node) in a network with no connection to the other nodes, indicating fragmentation in the population.

The **node degree** is the number of edges connecting to the node [40]. For example, node A and B in Figure 3 both have a degree of four, since there are four edges connecting to those nodes. Degree is used to identify nodes with a lot of connections to other nodes, used to model, for instance, disease spreading. The higher the degree, the more connections the node has [31].

2.3 Practical examples using animal social network metrics

An overview of the used metrics is shown in Table 3. Figure 4 is taken from *Snijders et al.* [2] and shows examples how different network metrics from SNA can be used for the monitoring of populations. The left part shows social networks and relevant questions related to wildlife conservation and management. The right part shows the quantification of those questions using social network metrics [2].

Table 3: Social Network Indicators

Metric	Purpose	Indications	Source
Density	Measures the number of connections as propor- tion of the number of possible connections	Connectivity in a net- work (indicates strength of social integration)	[2], [31]
Community detection	Measures the number of communities and their membership in a popu- lation	Strength of social inte- gration inside the popu- lation	[2], [31]
Component detection	Measures the number of communities in a net- work that are entirely disconnected	Amount of disconnected groups in a network, an indication of population fragmentation	[2]
Betweenness centrality	Indication for flow poten- tial (e.g. of informa- tion/diseases) between individuals or communi- ties in a network	The importance of indi- viduals in a network	[2], [31]
Degree	Measures the number of edges attached to a node	Insight in disease spreading	[31]

Animals are known to form territories around resources. Example **A** shows a situation with a high graph density. In this case resources are clumped together, resulting in the situation that animals from different territories are closely concentrated near each other, leading to aggressive interactions among individuals. In finding a solution to this, SNA can be used to indicate if the redistribution of resources will be effective. The redistribution lowers the graph density and increases the amount of components, decreasing the amount of aggressive interactions [2].

In conservation projects, groups of animals are regularly relocated (example **B**). SNA can be used to compare the group structure before and after relocation for evaluation purposes. Example **C** shows a situation where an animal is (illegally) killed by humans, causing fragmentation in the social group. SNA is used to understand if this is a temporary or permanent fragmentation and might help to predict what happens to the population when certain individuals disappear from the social network [2].

In example **D**, a group of animals have adjusted their behaviour due to environmental disturbances, changing at what time of the day the animals gather food. If due to these changes the animals shift their food gathering moment from daytime to the night, the group size likely increases. While this increases the safety, it might also cause more social conflict. SNA can be used for monitoring the long term situation [2].

Betweenness centrality is used in example **E** to identify individuals in a network that are important for the connectivity. Since these individuals are likely to spread diseases, they should be vaccinated. In **F**, SNA is used to identify central individuals that are essential to maintain social stability in the population. Removing those individuals might lead to instability in the group [2].

2.3.1 Other evaluation metrics

There are more evaluation metrics used, apart from the graph theory metrics. *Normalised mutual information* and *mean absolute deviation*.

Normalised mutual information Normalised mutual information (NMI) is used to validate that the communities found in the estimated graph are the same as in the real graph. Figure 5 illustrates the concept of mutual information, where I(X : Y) is the mutual information and H(X|Y) and H(Y|X) represent the non-mutual information (or variation of information), H(X) is the information in X where H(Y) is the information in Y. The mutual information I(X : Y) is defined in equation 3 [47].

$$I(X:Y) = \frac{1}{2}[H(X) - H(X|Y) + H(Y) + H(Y|X)]$$
(3)

Equation 3 is divided by two because H(X) is defined as I(X : Y) + H(X|Y) and H(Y) is defined as I(X : Y) + H(Y|X), which in theory should be the same, but because of possible errors the average of the two is used. The authors of [48] use the concept of

normalised mutual information to evaluate community structures using equation 4. Their algorithm creates a confusion matrix N, with rows corresponding to real communities the columns corresponding to the found communities. Matrix N_{ij} represents the nodes in the real community *i* that are present in the estimated community *j*.

$$I(A,B) = \frac{-2\sum_{i=1}^{c_A}\sum_{j=1}^{c_B}N_{ij}log(N_{ij}N/N_iN_j)}{\sum_{i=1}^{c_A}N_ilog(N_i/N) + \sum_{j=1}^{c_B}N_jlog(N_j/N)}$$
(4)

 c_A is the number of actual communities and c_B is the number of estimated communities. N_i is the sum over row *i* of matrix N_{ij} and N_j is the sum over column *j* of the matrix. *I*(*A*,*B*) is 1 when the two partitions (i.e. the real communities and the estimated communities) are identical to each other. When both partitions are totally independent of each other, *I*(*A*,*B*) equals 0 [48].

Mean absolute deviation The Mean Absolute Deviation (MAD) is the average distance between each datapoint in the set and the mean, giving an idea of the variability. The MAD value is calculated by calculating the mean of the dataset, than calculating the absolute deviation (the absolute distance to the mean) of each datapoint, summing the deviations and dividing them by the number of datapoints, as shown in formula 5.

$$MAD = \frac{\sum |x_i - \bar{x}|}{n} \tag{5}$$

2.4 Wireless technologies

Wireless communication is nowadays all around us, with WiFi and Bluetooth as well-known examples and are used for data communication between devices. In this project, wireless communication is used to check if nodes are within range of each other, indicating social interaction. Because a biotelemetry tag cannot easily be recharged while equipped on the animal, the tag should be energy efficient. For this reason, BLE and LoRa are used for the wireless communication between nodes [11].

2.4.1 Bluetooth Low Energy Overview

Bluetooth Low Energy (BLE, also known as Bluetooth Smart) operates in the 2.4 GHz industrial, scientific and medical (ISM) band and uses 40 (3 advertising and 37 data) channels, spaced 2MHz apart and is designed to be low-power. BLE devices can either be a transmitter, receiver or both (a transceiver) [49]. The developers of Bluetooth claim that a range of over 350 meters is possible [50].

Devices connected via a Bluetooth connection form a *piconet* operating in a master/slave configuration where the master initiates the connections. A master device is able to manage seven active connections simultaneously and can have 255 inactive ("parked") slave devices. The master can decide when certain slave devices are active or inactive, depending on the need to communicate. In BLE, four specific roles are specified, the *Broadcaster, Observer*,

Peripheral and *Central* roles. The *Broadcaster* role does not support connections and is used for data broadcasting (e.g. a thermometer broadcasting the current temperature). The *Observer* role complements the broadcaster, is designed for receiver only applications (e.g. receiving the broadcasted thermometer data) and does not support connections. Devices using the *Peripheral* role are optimised for a single connection, acts as a slave, need a Central device to connect to and cannot initiate connections on its own. The *Central* role supports multiple connections with devices in the Peripheral role, acts as a master and initiates connections. Several roles can be implemented in a device simultaneously [51].

In July 2017, *Bluetooth Mesh* was released in the form of a software update for existing Bluetooth devices supporting Bluetooth 4.0 or higher and is implemented as a flooding based mesh topology. This enables many-to-many device communication and is optimised for large-scale device networks [52]. All nodes in a mesh network must be a transceiver and can have three features. The *relay* feature re-transmits received messages to its neighbours, the *low power* feature is used by nodes to minimise the power consumption by going to sleep and periodically wakes up to receive messages. For this, the node must form a friendship with a neighbouring node which has the *friend* feature enabled. The low power node will become dependent on the friend node, who will become a cache for the low power node. The cached messages will be send when the low power node wakes up [53].

2.4.2 Long Range (LoRa) Overview

LoRa (Long Range) is designed to provide long-range, low-power wireless data communication operating in Europe on the 868 MHz ISM band with a data rate of up to 50 Kbps. A LoRa network is designed to be used as a star of stars topology consisting of *LoRa End Devices* and *LoRa Gateways*. End devices are typically sensor nodes communicating with the gateways acting as relays, which on their turn forward the received data to network servers [54]. Empirical tests show that about 80% of packages send by nodes successfully reach the base station on a distance of up to 5km, where more than 60% of the packages successfully arrive at the base stations on a distance between 5 and 10km. On distances greater than 10km the majority of packages does not arrive at the base station [55]. Even though LoRa is designed to be a star of stars topology, it is also possible to let nodes communicate with each other in a mesh topology [56].

LoRa uses orthogonal spreading factors, ranging from spreading factor 7 to 12, enabling multiple spread signals to be transmitted at the same time and on the same channel. Table 4 shows the spreading factors, their corresponding bit rate and the transmit power [57]. The higher the spreading factor, the further the range, but at the cost of a lower bit rate [58].

2.4.3 Path loss

Path loss can be caused by five different mechanisms, at first due to **Free-space propa**gation, when there are no obstructions the transmitted signal strength decreases over dis-

Spreading factor	Bit rate (bit/s)	TX Power (dBm)
SF12	250	20
SF11	440	14
SF10	980	11
SF9	1760	8
SF8	3125	5
SF7	5470	2

Table 4: LoRa spreading factors with bit rate and transmission (TX) power

tance, where the signal strength decreases by a factor of $1/d^n$, where *d* is the distance and *n* is the path loss exponent. The signal strength logarithmically decreases with the distance with 10nlog(d) [49].

Path loss is heavily influenced by the environment, where objects can possibly block the radio signal, weaken the signal or change its direction. Those effects are **transmission influences**, where the signal penetrates trough a medium (e.g. a concrete wall) resulting in a loss of signal strength, **reflections**, which happens when the signal waves impinge upon big surfaces (e.g. buildings). Depending on how the signal reflects, the range can be decreased or increased. **Diffraction** happens when a signal is obstructed by an object with sharp edges, creating secondary waves. The last mechanism is **scattering**, where the signal interacts with a large number of small objects (e.g. on foliage, street signs), scattering the signal in multiple directions [49]. Due to such phenomena, received signal strength is not suitable for exact localisation, but only for location/distance estimation [59]. For the modelling of these phenomena, path loss models are developed.

2.4.4 Path loss models

The used path loss models in the simulation are discussed below, the *Simplified path loss model* for BLE and the *Hata-COST model* for LoRa.

Simplified Path Loss Model

The simplified path loss model is a path loss model aiming to cover the general signal propagation without the need of using complex path loss models [1]. Equation 6 shows the simplified path loss model, with the variables explained in 5, equation 7 shows how K is computed [1] and Table 6 shows an overview of the path loss exponents [49].

$$P_r dBm = P_t dBm + K dB - 10 \log_{10} \gamma[\frac{d}{d_0}]$$
(6)

$$K(dB) = -20 log_{10}(4\pi d_0/\lambda)$$
(7)

where K is dependent of the antenna characteristics and average channel attenuation (SPECIFY), d₀ is a reference distance for the antenna far-field (SPECIFY) and γ is the path

Symbol	Description
P_t	Transmit power
P_r	Received power
γ	Path loss exponent
K	Unitless constant for attenuation and antenna characteristics
d	Distance
d_0	Far-field reference distance

Table 5: Variables for simplified path loss models

Table 6: Path loss exponents for different environments

Environment	Path loss exponent (γ)
Free space	2
Urban area	2.7 - 3.5
Shadowed cellular radio	3 - 5
Line-of-sight in building	1.6 - 1.8
Obstructed in building	4 - 6
Obstructed in factories	2 - 3

loss exponent (see table 6) [49].

Okumura-Hata model

The Okumura-Hata model is considered to be one of the most accurate models, capable of accurately estimating path loss in different environments. It is an empirical model, originally based on the Tokyo area and refined by Hata. Equation 8 is the equation for the predicted path loss in an urban environment. Table 7 discusses the variables [49]. The correction factor $A(h_r)$ for a small or medium-sized city is given by equation 9, while equation 10 and equation 11 are the correction factors for a large city, with a $f_c \leq 300MHz$ and $f_c \geq 300Mhz$, respectively [49].

$$L_{dB}(small/mediumcity) = 69.55 + 26.16 \log(f_c) - 13.82 \log(h_t) - A(h_r) + (44.9 - 6.55 \log(h_t)) \log(d)$$
(8)

Variable	Details
F_c	Carrier frequency in MHz from 150 to 1500 MHz
H_t	Height of transmitting antenna (base station) in meters, from 30 to 300m
H_r	Height of receiving antenna (mobile unit) in meters, from 1 to 10 m
d	Propagation distance between antennas in km, up to 20 km
$A(h_r)$	Correction factor for mobile unit antenna height

Table 7: Varia	ables for Oku	mura-Hata model
----------------	---------------	-----------------

$$A(H_r) = 1.1\log(f_c) - 0.7)h_t - (1.56\log(f_c) - 0.8)dB$$
(9)

$$A(H_r) = 8.29[log(1.54h_r)]^2 - 1.1dB$$
(10)

$$A(H_r) = 3.2[log(11.75h_r)]^2 - 4.97dB$$
⁽¹¹⁾

For the path loss in a suburban environment or an open environment, 12 and 13 are used [49].

$$L_{dB}(suburban) = L_{dB}(small/mediumcity) - 2[log(f_c/28)]^2 - 5.4$$
 (12)

$$L_{dB}(rural) = L_{dB}(small/mediumcity) - 4.78(log f_c/)^2 - 18.733(log f_c) - 40.98$$
(13)

Hata-COST Model

The Hata-COST model is developed by the European cooperative for scientific and technical research (EURO-COST) and extends the Okumura-Hata model to 2 GHz [1]. The model is widely used because of its simplicity and availability of correction factors for different types of environments. Equation 14 shows the path loss calculation for urban areas, equation 15 is the equation for suburban areas. The used variables are the same as in Table 7, with the exception of C_m , which is 0dB for medium-size cities and suburbs and 3dB for metropolitan areas [1].

$$P_{L,urban}(d)dB = 46.3 + 33.9 \log_{10}(f_c) - 13.82 \log_{10}(h_t) - a(h_r) + (44.9 - 6.55 \log_{10}(h_t)) \log_{10}(d) + C_{max}(14)$$
(14)

$$P_{L,suburban}(d) = P_{L,urban}(d)dB - 2[log(f_c/28)]^2 - 5.4$$
(15)

2.4.5 Log-normal shadowing

Path loss is not the only influence on the range of the transmitted signal. Random variations caused by objects in the signal path can cause changes at the path loss at given distances. The signal can also reflect on surfaces, causing scattering, which should also be taken into account, as also discussed in section 2.4.3. Log-normal shadowing is the most common model for this and it is empirically confirmed that the model accurately models the variation in path loss or received power in both outdoor and indoor environments. Without the shadowing taken into account in the simulator, the range of a node will be a perfect circle, which is not a realistic scenario. Figure 6a shows the range when only path loss is used and when path loss is combined with shadowing. Figure 6b shows this effect on a linear scale [1].





(b) Path loss vs coverage, with and without shadowing

(a) Estimated path loss and estimated path loss combined with shadowing, with radius R and base station BS



2.5 Summary

Social network analysis has made a couple of notable advances in the past hundred years (due to the formalisation of fundamental concepts, studies to network structures and social ties). Eventually leading to online social networks and a thorough understanding of human social network analysis, leading to the adoption of social network analysis in different fields, including psychology, economy and biology where the latter one uses social network analysis on animals, animal SNA.

To be able to perform animal SNA, data is needed which can be gathered in various ways. Methods of data gathering are using camera traps, acoustic fixed arrays, mark- and recapture, biologging and biotelemetry. The gathered data can then be used for the analysis of animal networks, where used metrics are density, community detection, component detection, betweenness centrality and the node degree.

BLE and LoRa are suitable wireless technologies for animal tags because both techniques are designed for a low energy consumption. The challenge in deriving the distance between nodes by using the received signal strength is that the signal strength is not a true representation of the actual distance, but rather an estimation. The signal strength is heavily influenced by the environment, signals can for instance be blocked, weakened or scatterred by objects in the environment.



Figure 4: Examples of social network analysis usage in wildlife conservation (image taken from Snijders et al. [2])



Figure 5: Mutual information where I(X : Y) represents the mutual information and H(X|Y) and H(Y|X) the non-mutual information

3 METHODOLOGY

3 Methodology

This chapter discuss how the simulation model works, section 3.1, the experimental method, section 3.2, and the evaluation method, section 3.3.

3.1 Simulation method



Figure 7: Overview of the system simulation tool

Figure 7 is an overview of how the simulation tool works. A simulation dataset is used as input and uses the x,y coordinates of nodes to calculate the euclidean distance between all possible node pairs, per timeframe, and stores this in a matrix called the *actual distance matrix* (an example with four nodes can be seen in Table 8).

	1	2	3	4
1	0	700	40	190
2	700	0	320	430
3	40	320	0	70
4	190	430	70	0

Table 8: Example of the actual distance matrix with four nodes

The actual distance matrix is used as input to calculate the estimated distance between

nodes, by using the actual distance in combination with a path loss model and log-normal shadowing. This results in a matrix containing the expected path loss between all possible pairs of nodes, which in turn is used to recalculate the distance between all possible nodes. This results in a matrix with the estimated distances, the *estimated distance matrix*, as seen in Table 9.

	1	2	3	4
1	0	680	45	205
2	680	0	310	415
3	45	310	0	85
4	205	415	85	0

Table 9: Example of the estimated distance matrix with four nodes

The actual distance and estimated distance matrices are used to check whether two nodes are in range of each other, by checking the distance between nodes (using the actual distance matrix for the actual network and the estimated distance matrix for the estimated network) and comparing this to the threshold, which is the theoretical range of the radio channel (from BLE or LoRa). If the distance between the nodes is less than the threshold, the nodes are assumed to be in range of each other and get an edge (represented by a 1) in their corresponding adjacency matrices. This can be seen in Table 10 for the actual distance matrix and in Table 11 for the estimated distance matrix. The two adjacency matrices are calculated for every time frame.

The *actual distance matrix* and the *estimated distance matrix* are used as input for the data evaluation. The method used for the data evaluation is discussed in section 3.3.

Table 10: Example of an adjacency matrix based on the actual distance matrix (Table 8) for 4 node pairs

	1	2	3	4
1	0	0	1	1
2	0	0	0	0
3	1	0	0	1
4	1	0	1	0

Table 11: Example of an adjacency matrix based on the estimated distance matrix (Table 9) for 4 node pairs

	1	2	3	4
1	0	0	1	0
2	0	0	0	0
3	1	0	0	1
4	0	0	1	0

3 METHODOLOGY



(a) Photo 1 of the BLE test location

(b) Photo 2 of the BLE test location

Figure 8: Photos of the BLE test location

3.1.1 Used path loss models

The used path loss models are the *simplified path loss model* for BLE and the *Hata COST model* for LoRa. Both models were chosen for their accuracy and simplicity, as discussed below.

BLE

According to the simplified path loss model, the maximum range of BLE in an open/rural area is 199.46 meters, with the parameters used as in Table 12. To validate this, two Nordic NRF52 devices were programmed to connect with each other using BLE. The sensitivity of the devices is -93dB and the transmission power is 4dBm, corresponding to the simulation model. The range was then tested in a park, an area with a field, trees and people walking with dogs, as can be seen in Figures 8a and 8b. Figure 9 shows the area from the air.



Figure 9: Overview of the test location, a and b correspond to the place the photos from Figure 8 were taken. Image taken from Google Maps

The findings are that the devices easily stayed connected when the distance between

3 METHODOLOGY

the two devices was approximately 170 meters. After this, the devices subsequently disconnected but were able to reconnect again. The devices showed serious connection problems when at a distance of approximately 180 meters, but sometimes was able to reconnect again, resulting in a maximum distance of about 190 meters. At 190 meters, no connection was possible. The devices were held at chest height.

LoRa

For LoRa, the *simplified path loss model, Okumura-Hata model* and the *Hata-COST model* were considered. Figure 10 shows the estimated range of LoRa calculated by using three path loss models. As can be seen, both the Okumura Hata model and the Hata-COST model are almost the same, about 1,5km with a transmission power of 2dBm which is expected [11], but the simplified path loss model is very off, probably because the use of only a path loss exponent is sufficient for transmissions over shorter distances, like BLE, but is not suitable for larger ranges, so an empirical model is more suitable. Because of its simplicity and accuracy, the Hata-COST model is used as the path loss model for LoRa. The Hata-COST model gives a maximum range of approximately 1.5 km when using the parameters in Table 13, corresponding to the information in [11]. The shadowing is simulated by using log-normal shadowing with a mean of 0 and a standard deviation of 4. The environment is assumed to be rural and spreading factor 7 is used.



Figure 10: Comparison of the simplified path loss model, the Okumura-Hata model and the Hata-COST model

Two nodes were configured to test the practical range of LoRa. One node was programmed to broadcast a message every second using LoRa and the other nodes was


(a) Photo 1 of the LoRa test location



(c) Photo 3 of the LoRa test location



(b) Photo 2 of the LoRa test location



(d) Photo 4 of the LoRa test location



configured to listen for broadcasts, both using spreading factor 7. The nodes were also programmed to save their GPS location every second. The used hardware is described in section 5.1. The range test was performed near the Aamsveen, a fen area with rural surroundings. The area contains a road surrounded by big open fields, sometimes a house or farm and (big groups of) trees. Figure 11 shows four pictures of the area.

One node was placed in the yard of a house located in the area, the other node was mounted on a bike that was used to travel trough the area. After 1.2km the receiving node received a broadcast message approximately once every five seconds instead of every second. After 1.35km the node stopped receiving broadcasts. The difference between the practical and theoretical range might be caused by the big groups of trees that are located in the area, possibly (partly) blocking the signal or causing scattering effects.

3.2 **Experimental design**

For the experiments, nodes are programmed to broadcast their node id using BLE and LoRa and to scan for devices in the neighbourhood once a minute, using both BLE and LoRa. It also saves the node's GPS location once a minute, during Bluetooth and LoRa scanning. For data gathering purposes, nodes are placed according to certain scenarios. This is discussed in more detail in section 5.1.

The results of the experiments are an actual and an estimated adjacency matrix for each scenario. Those are used as input in the evaluation model. The method used for the data evaluation is discussed in section 3.3.

3.3 Evaluation method and algorithms

For the data evaluation, the actual and estimated adjacency matrices are used to create two network graphs, as seen in Figure 12. The *actual network*, based on the actual adjacency matrix and the *estimated network*, based on the estimated adjacency matrix. Both networks are used to compare the real social graph with the estimated social graph using the metrics found in the literature study, as described in section *2.2 animal social network data analysis*. The network is checked for isolated nodes to find out if nodes are accidentally missed and isomorphism is used to check if there are cases where the actual network is identical to the estimated network. If both networks are identical, it would mean that the social network analysis of both networks will give exactly the same results. The method and algorithms used for evaluation is discussed in more detail in section 3.3.1.



Figure 12: Overview of the evaluation design

3.3.1 Metrics and procedure

This section discusses the metrics used for the social network analysis and how the metrics are validated, including the algorithm.

Betweenness

The literature shows that nodes with a high betweenness are nodes of interest in an ASN. However, the definition of "high" is not defined in the literature, so the choice has been made to define nodes as high when the betweenness of a node is 0.85 times the highest cost in the graph *and* the cost is above average, where the higher the cost, the more important a node is. Algorithm 1 shows the algorithm for one timeframe.

```
Result: High betweenness check (for one timeframe)
G = graph actual distances;
GE = graph estimated distances;
BetweennessG = centrality(G,'betweenness',Cost,G.Edges.Weight);
BetweennessGE = centrality(GE,'betweenness',Cost,GE.Edges.Weight);
SumBetweennessG = sum(BetweennessG);
SumBetweennessGE = sum(BetweennessGE);
MeanBetweennessG = mean(BetweennessG);
MeanBetweennessGE = mean(BetweennessGE);
HighBetweennessG = max(BetweennessG)*0.85;
HighBetweennessGE = max(BetweennessGE)*0.85;
NumHighBetweennessNodesG = 0;
NumHighBetweennessNodesGE = 0;
i = 1;
while i smaller than 51 do
   if G.betweenness(i) > MeanBetweennessG & G.betweenness(i) >
    HighBetweennessG then
      NumHighBetweennessNodesG = NumHighBetweennessNodesG + 1;
   end
   if GE.betweenness(i) > MeanBetweennessGE & GE.betweenness(i) >
    HighBetweennessGE then
      NumHighBetweennessNodesGE = NumHighBetweennessNodesGE + 1;
   end
   i = i + 1;
end
          Algorithm 1: High betweenness check (for one timeframe)
```

A couple of validation tests are performed to check if the actual network correspond with the estimated distance. The first validation checks if the node with the highest betweenness value in both graphs is the same node. The second validation test is to see if there are overlapping nodes in the graphs, resulting in a score between 0 and 1, calculated by dividing the amount of the same nodes with a high betweenness by the amount of total nodes with a high betweenness in the graph based on the actual distances. For example, if nodes 12 and 15 are high nodes in the actual network, and nodes 12 and 18 are high nodes in the score is 1 overlapping node over 2 total nodes, resulting in a score of 0.5. The last validations checks if the amount of nodes with a high score is the same in both networks. If not, it checks what the offset is.

Node degree

Similar as for betweenness, the approach is to identify, per timeframe, the nodes with a high degree in the actual and estimated network. Again, nodes with a high degree are defined when their degree is 0.85 times the highest degree in the graph *and* when their

3 METHODOLOGY

degree is above average. This is covered in algorithm 2. For the validation, the amount of high degree nodes per timeframe are checked in both networks, as well as the amount of similar nodes.

```
Result: High degree check (for one timeframe)
G = graph actual distances;
GE = graph estimated distances:
DegreeG = centrality(G,'degree', Importance, G.Edges.Weight);
DegreeGE = centrality(GE,'degree',Importance,GE.Edges.Weight);
SumDegreeG = sum(DegreeG);
SumDegreeGE = sum(DegreeGE);
MeanDegreeG = mean(DegreeG);
MeanDegreeGE = mean(DegreeGE);
HighDegreeG = max(DegreeG)*0.85;
HighDegreeGE = max(DegreeGE)*0.85;
NumHighDegreeNodesG = 0;
NumHighDegreeNodesGE = 0;
i = 1;
while i smaller than 51 do
   if G.Degree(i) > MeanDegreeG & G.Degree(i) \geq HighDegreeG then
      NumHighDegreeNodesG = NumHighDegreeNodesG + 1;
   end
   if GE.Degree(i) > MeanDegreeGE & GE.Degree(i) ≥ HighDegreeGE then
      NumHighDegreeNodesGE = NumHigDegreeNodesGE + 1;
   end
   i = i + 1;
end
```

Algorithm 2: High degree check (for one timeframe)

Density

Equation 1 is used to calculate the density in both graphs per timeframe, as can be seen in algorithm 3. For the evaluation, the difference between the actual and estimated density value is calculated per timeframe, defined by DensityDifference = ActualDensity - EstimatedDensity.

Component detection

A component in a network is defined as a group in a network, with at least more than one node, with no connections to other groups. Component detection is used to measure the number of components, showing signs of fragmentation in the population. As an example, when measuring the connected components in Figure 3, the graph shows one component (because of the connection between nodes A and B). If nodes A and B would not have an

3 METHODOLOGY

Result: Density calculation (for one timeframe) G = graph actual distances; GE = graph estimated distances; NumEdgeG = numedges(G); NumEdgeGE = numedges(GE); NumNodesG = numnodes(G); NumNodesGE = numnodes(GE); DensityG = (2*NumEdgeG) / (NumNodesG*(NumNodesG - 1)); DensityGE = (2*NumEdgeGE) / (NumNodesGE*(NumNodesGE - 1)); Algorithm 3: Density calculation (for one timeframe)

edge, the figure would contain two components.

The validation method counts the number of components and the number of isolated nodes in both networks and subtracts the number of isolated nodes from the number of components (because the goal is to find components with at least two members).

```
Result: Component detection (for one timeframe)
G = graph real locations;
GE = graph estimated locations;
ConnCompG = conncomp(G);
ConnCompGE = conncomp(GE);
NumIsolatedNodesG = 0:
NumIsolatedNodesGE = 0;
i = 1;
while i smaller than 51 do
   if G.degree(i) == 0 then
      NumIsolatedNodesG = NumIsolatedNodesG + 1;
   end
   if GE.degree(i) == 0 then
      NumIsolatedNodesGE = NumIsolatedNodesGE + 1;
   end
   i = i + 1:
end
NumSubgroupsG = (ConnCompG - NumIsolatedNodesG) NumSubgroupsGE =
 (ConnCompGE - NumIsolatedNodesGE)
            Algorithm 4: Component detection (for one timeframe)
```

For the evaluation, it is checked if the amount of components are the same in both graphs. If not, it checks what the offset is. For example, if the actual network has three components, but the estimated network contains two components, the offset is -1.

Communities

For the detection of the optimal community structure, each entry in the dataset is checked for its optimal number of communities (k) based on its NMI value (Q) which has the number of connected components (stored in vectors Vtemp and VEtemp) as input, as can be seen in algorithm 5. Since every timeframe in the dataset contains 50 nodes, it is possible that one timeframe has a maximum of 50 different communities.

```
Result: Community structure detection
G = graph real locations;
GE = graph estimated locations;
Q = 0;
Qtemp = 0;
k = 0;
i = 1:
while i smaller than 51 do
   commdetreal = GirvanNewman(G,i);
   commdetEstimated = GirvanNewman(GE,i);
   Vtemp = connectedcomponents(commdetreal);
   VEtemp = connectedcomponents(commdetEstimated);
   Qtemp = NMI(Vtemp,VEtemp);
   if Qtemp bigger than Q then
      Q = Qtemp;
   end
   i = i + 1;
end
                      Algorithm 5: Community detection
```

Isolated nodes

To detect the number of nodes per timeframe that are completely isolated from the rest of the network, the node degree has to be checked. If the degree of a node is 0, the node is completely disconnected from the rest of the network [40]. The used algorithm checks the number of nodes with a degree of 0 in the actual network and the estimated network, as can be seen in algorithm 6.

For the evaluation, it is checked if the amount of isolated nodes is the same in both graphs. If not, their offset is calculated. If the amount of isolated nodes in the actual network is 0 and 1 in the estimated network, the offset is +1. It also checked if the isolated nodes found are the same, resulting in a score between 0 and 1. If there are no isolated nodes in both graphs, the score is 1. If there are 0 isolated nodes in the graph based on actual distances, but at least one in the graph based on the estimated distances, the score is 0.

Algorithm 6: Isolated node calculation (for one timeframe)

Isomorphism

To check if the two networks in each timeframe are isomorph, the *isisomorphic* function is used, which returns a 1 when the two graphs are isomorph, otherwise it returns a 0. Algorithm 7 shows the algorithm for one timeframe. The number of timeframes where the two graphs were isomorphic is used for validation.

```
Result: Isomorphism check (for one timeframe)
```

```
G = graph actual distances;

GE = graph estimated distances;

i = 1;

while i smaller than 51 do

| IsomorphismCheck = isisomorphic(G,GE);

i = i + 1;

end
```

Algorithm 7: Isomorphism check (for one timeframe)

4 Simulation setup, results and evaluation

This section discusses the results and evaluates them.

4.1 Used simulation datasets

The simulation has two datasets as input (as seen in the data section), with each dataset existing of 8000 timeframes. Every timeframe contains the location of 50 nodes, representing animals. The simulation uses this data to calculate the *actual distance* between nodes, which is the Euclidean distance (a straight-line distance between two points) between all the nodes. This is done for all the timeframes. The **actual** distance is calculated for two purposes. At first to check if the nodes are within the critical distance of each other, i.e. the theoretical maximum distance where two nodes are still able to establish a radio link (calculated using a path loss model and shadowing). The second purpose is that the **actual** distance between nodes is used for the calculation of the *estimated distances* between nodes. This **estimated** distance is inferred from the actual distance by using a pathloss model in combination with lognormal shadowing, used to convert the actual distance to the theoretical received signal strength. Converting this received signal strength to distance gives the estimated distance between nodes.

4.2 Path loss model setup

Tables 12 and 13 shows the values corresponding to the used parameters for the simplified path loss model, for BLE, and the Hata-COST model, for LoRa, respectively.

Symbol	Description
P_t	2dBm
Receiversensitivity	-93dBm
γ	2.7
d_0	1m
$\sigma_{\psi_{dB}}$	3.65
f_c	2.4GHz

Table 12: Parameter values used in the simulation for BLE

4.3 Simulation results and evaluation

This section discusses the results of the simulation and evaluates them.

4.3.1 Isomorphishm

None of the graphs were isomorph, indicating that there is not a single timeframe where the actual network and the estimated network are exactly the same. Since isomorphism gives a

Symbol	Description
P_t	2dBm
Receiversensitivity	-123dBm
h_t	50m
h_r	2m
f_c	868MHz
C_m	0

Table 13: Parameter values used in the simulation for LoRa

binary output (two graphs are exactly the same or two graphs are different), it is unclear to what extend the graphs differ. It might be that only one edge is different but there is also a possibility that both graphs are completely different and have no similarity at all. The metrics below are used to give an insight in how the graphs differ from each other.

4.3.2 Isolated node detection

Figure 13 shows the offset of the found number of isolated nodes found in the actual network and the estimated network. For BLE, it can be seen that in 68% of the timeframes the amount of isolated nodes is the same in both graphs, where this is 40.1% for LoRa. In 16.6% and 11.4% of the timeframes the offset is, respectively one isolated node more or less for BLE, where this is 22% and 18.2% for LoRa, respectively. In 2.2% and 1.6% of the time there were two more or two less isolated nodes detected for BLE while this is 8.2% and 5.9% for LoRa. For BLE, in 40% of the timeframes

Figure 14 shows the same node score for the isolated nodes, where timeframes without isolated nodes are omitted. It shows that for BLE, in 22.6% of the timeframes the similarity score is 1, for LoRa this is 20.9%. For BLE the score is 0 in 72.1% of the timeframes, against 67.2% for LoRa. There was at least one isolated node in the estimated graph while there was no isolated node in the actual graph in 12% of the timeframes and no isolated nodes in both graphs in 57.2% of the timeframes for BLE, for LoRa this was 8% and 40.4%, respectively.

BLE performs a bit better on the isolated node detection, especially by identifying the correct number of isolated nodes in the graph, suggesting that BLE is less likely to miss connections between nodes than LoRa does. It should be taken into account that there is a chance for both techniques to miss or falsely identify a link between nodes, resulting in a node that appears to be disconnected from the network, but in reality is or vice versa. A reason for this can be that, even though two animals are in theoretical proximity of each other, the BLE or LoRa signal is not detected between nodes due to, for instance, trees blocking the signal. On the other hand, the signal can be transmitted further than expected due to reflections, leading to the situation that, even though a node is in reality isolated, other nodes are still able to receive the broadcast of the isolated nodes.



Figure 13: Simulation results, isolated nodes: Offset of the number of isolated nodes in the actual graph compared to the estimated graph

Both technologies seem to be suitable for the detection of nodes in a network, but if BLE or LoRa incorrectly misses or identifies a node as isolated, it is hard to identify which exact node it is.

4.3.3 Betweenness

Figure 15 shows the offset of nodes with high betweenness, compared to the actual graph based on the actual and estimated graph, showing that the amount of nodes is the same in most timeframes (31.9% for BLE, 31.1% for LoRa) and that offsets of 1 node are also common (-1 is 21.1%, +1 is 23.3% for BLE and 16.2% and 23.8% respectively for LoRa). There are almost no nodes with high betweenness that are the same in both graphs, with 82.6% of the nodes showing no similarity for BLE and 75.8% for LoRa, as can be seen in Figure 16.

BLE performs a little bit better on betweenness when looking at the amount of nodes, but both technologies are not suitable for identifying the exact amount of nodes with a high betweenness because the 0, -1 and +1 offset are quite close to each other. So the technologies are suitable for an estimation of the amount of nodes. Both BLE and LoRa are not suitable for identifying which exact nodes have a high betweenness. This is again probably due to transmission influences. If two animals are barely within or out of the theoretical



Figure 14: Simulation results, isolated nodes: Same node score

range between each other, the situation can occur where, an animal is incorrectly identified as nearby, or vice versa, while this is not true. This influences the node degree and thus the betweenness.

4.3.4 Node degree

Figure 17 shows the offset of nodes with a high degree compared to the actual graph and the estimated graph, showing that the amount of nodes with an above average degree is likely to differ between the real and estimated graphs. Figure 18 shows the overlapping node scores for the above average degree and high degree nodes. In most cases, less than 50% of the nodes in the estimated graph show overlap with the actual graph.



Figure 15: Simulation results, betweenness: Offset of the number of nodes with high betweenness in the actual graph compared to the estimated graph



Figure 17: Simulation results, degree: Offset of the number of nodes with high degree in the actual graph compared to the estimated graph



Figure 16: Simulation results, betweenness: Same node score



Figure 18: Simulation results, degree: Same node score

The node degree is very hard to determine, for both BLE and LoRa. This is probably due to path loss and shadowing phenomena. If animals are barely within or out of theoretical reach, the possibility exists that nodes can receive each others broadcast, while they are out of reach or vice versa. Also transmission blockage due to trees and other objects can influence the range of the broadcast. If all animals are close to each other, the degree is probably easier to determine than if all animals are scattered over the area. This is probably because shadowing and path loss have a higher influence when the transmission range is longer. Both BLE and LoRa are unsuitable for finding the amount of nodes with a high degree and unsuitable to find out which nodes have a high degree.

Detailed results for one timeframe

Figure 19 shows the actual network (a) and the estimated network (b) for one timeframe. The nodes are plotted in a random way and does not represent a node location. Table 14 compares the node degree of the nodes in both networks. A red number indicates that the estimated degree differs from the actual degree. As seen in the table, in this case only eleven nodes have the same actual and estimated degree. Appendix A contains the table corresponding to Figure 19a (Table 19) and the table corresponding to Figure 19b (Table 20).



(a) Actual network

(b) Estimated network



Node	Actual degree	Estimated degree	Node	Actual degree	Estimated degree
1	3	3	26	11	15
2	8	6	27	6	5
3	9	11	28	3	6
4	2	5	29	2	4
5	5	4	30	10	12
6	11	9	31	2	3
7	11	13	32	14	15
8	6	7	33	1	2
9	4	4	34	11	16
10	3	3	35	11	13
11	2	3	36	5	7
12	7	6	37	2	4
13	8	11	38	7	6
14	6	7	39	3	4
15	6	5	40	6	6
16	11	13	41	4	4
17	3	3	42	9	10
18	9	6	43	3	4
19	9	12	44	7	9
20	5	4	45	9	9
21	8	8	46	4	4
22	4	6	47	4	4
23	2	4	48	3	3
24	5	4	49	11	14
25	4	5	50	7	9

Table 14: Node degree comparison between the actual and estimated network corresponding to Figure 19

4.3.5 Density

For BLE, the average density in the actual graphs is 0.1125 with an MAD of 0.0058, the average density in the estimated graphs is 0.1245 with an MAD of 0.0065. For LoRa, the average density in the actual graphs is 0.0957 with an MAD of 0.0081, the average density in the estimated graphs is 0.1134 with an MAD of 0.0091.

Figure 20 shows the density difference between the actual density of a timeframe and the estimated density of the timeframe. As seen, in most timeframes the difference is between -0.001 and -0.0019 (46.6% for BLE and 50.4% for LoRa), followed by -0.02 and -0.029 (BLE 37.8% and LoRa 32.6%). A difference between 0 and -0.009 occurs in 10.3% of the timeframes for BLE and in 12.6% for LoRa.



Figure 20: Simulation results, density: Difference between the actual density and the estimated density

Since density is measured in a value between 0 and 1 and the measured difference between the actual and estimated density values is very small, the results are very good. Both BLE and LoRa are suitable for the calculation of the graph density, with LoRa performing a little bit better than BLE. It should be noted that the estimated density value is probably a bit higher than the actual density.

4.3.6 Component detection

Figure 21 shows that, for BLE, 15.2% of the timeframes has an offset of 0, while this is 32.6% for LoRa, an offset of +1 and +2 has occurred in 34.7% and 33% of the timeframes for BLE, respectively, and in 36.1% and 19.8% for LoRa. An offset of +3 and +4 is seen in 13.6% and 2.2% for BLE, respectively, and 7.2% and 2% for LoRa.



Figure 21: Simulation results, component detection: Offset of the number of components in the actual graph compared to the estimated graph

BLE performs a lot better than LoRa, but both technologies show a large offset in the amount of found components, probably because both BLE and LoRa miss (or in a few cases falsely identifies) important links between nodes, resulting in more components in a graph than there are in reality (or less in a few cases). Since component detection is used to give insight in the fragmentation of a network, using BLE and (especially) LoRa for component detection seems only useful when it is taken into account that the actual amount of isolated components is probably one or two less than predicted and only gives a rough estimation and not an exact amount. The cause for this is probably again that some animals are just out of or within the theoretical distance and due to signal blockage and/or path loss, links between (groups of) animals are missed. This results in the situation that components that are linked together in the actual network are sepperated in the estimated network.

Detailed result for one timeframe

Figure 22b shows the actual network graph and the estimated network graph for one timeframe, which is not a representation of node locations. The nodes are plotted in a random manner. The amount of components is two for the actual network and three for the estimated network. However, the latter one shows one disconnected node, which does not count as a component. So both graphs contain two components. An overview of which node is in which component can be found in Table 15. For validation purposes, the corresponding adjacency matrices can be found in Table 21 and Table 22 in appendix A.



(a) Found components based on the actual net-(b) Found components based on the estimated work network

Figure 22: Graph in the graph based on the actual distances (a) and in the graph based on estimated distances (b) for one timeframe

Actual		Estimated			Actual	Estimated	Estimated		
component 1		component 1			component 2	component 2	isolated nodes		
1	18	35	1	18	35	10	10	11	
2	19	36	2	19	36	11			
3	20	38	3	20	38	23	23		
4	21	39	4	21	39	33	33		
5	22	40	5	22	40	37	37		
6	24	41	6	24	41				
7	25	42	7	25	42				
8	26	43	8	26	43				
9	27	44	9	27	44				
12	28	45	12	28	45				
13	29	46	13	29	46				
14	30	47	14	30	47				
15	31	48	15	31	48				
16	32	49	16	32	49				
17	34	50	17	34	50				

Table 15: Component comparison between the actual and estimated graph

4.3.7 Communities

The results show that the average NMI value is 0.8846 for BLE and 0.8324 for LoRa. Most entries (51.5% for BLE, 49.8% for LoRa) score an NMI value between 0.80 and 0.89. For BLE, 9.28% of the entries scores the maximum NMI value of 1, for LoRa this is 5.52%. A score between 0.90 and 0.99 is obtained in 29.76% of the timeframes for BLE and 9.88% for LoRa. Finally, 9.44% (BLE) and 34.9% (LoRa) obtained a score between 0.70 and 0.79, as can be seen in figure 23.



Figure 23: Simulation results, community detection: NMI values

The results show that BLE performs a lot better in the community detection algorithm than LoRa. This can be explained by the same reasoning as earlier. Both techniques miss important links between nodes due to shadowing and path loss phenomena, resulting in a loss of network information. Especially LoRa suffers from this, it probably suffers more from shadowing and path loss because of the far distance between nodes. Still, both technologies show quite good results and seem to be useful for the estimation of community structures. Information about the exact community structure is not likely to be achieved.

Detailed result for two timeframes

The community evaluation combined with NMI is explained by discussing the community detection in two different examples. In the first example comparing the actual communities (inferred from the actual network) and estimated communities (inferred from the estimated network) with an NMI of 1. The actual and estimated communities were found in the same timeframe. In the second example, the actual and estimated communities have an NMI value of 0.95. Again, the actual and estimated communities were found in the same timeframe. This is visualised by showing the graphs in a graphical way, plotted in a figure. It should be noted that the place where nodes are plotted is in a random way and do not represent locations.





(b) Found communities based on the estimated network

(a) Found communities based on the actual network

Figure 24: Communities found in the graph based on the actual network (a) and in the graph based on the estimated network (b) for one timeframe



(a) Found communities based on the actual network (b) Found communities based on the estimated network

Figure 25: Communities found in the graph based on the actual network (a) and in the graph based on the estimated network (b) for one timeframe

Figure 24 shows the actual communities found (Figure 24b) based on the actual network and the estimated communities found, based on the estimated network (Figure 24a). Tables 23 and 24 in appendix B show the corresponding adjacency matrices for the actual and estimated communities, respectively. Calculating the NMI gives a value of 1 for the two found communities, which indicates that both communities are the same, as can be seen in Table 16.

Figure 25 shows the found actual communities (Figure 25a) based on the actual network and the estimated communities, based on the estimated network (Figure 25b). Tables 25 and 26 in appendix B show the corresponding adjacency matrices for the actual and estimated communities, respectively.

Calculating the NMI for the two communities gives a value of 0.95, which indicates that both communities are mostly alike, but not completely the same. Table 17 shows that in the actual communities, node 26 is a member of community 2 but is found in community 3 in the estimated communities.

Г

Communities										
Actual 1	Estimated 1	Actual 2	Estimated 2	Actual 3	Estimated 3					
1	1	2	2	9	9					
4	4	3	3	10	10					
12	12	5	5	11	11					
14	14	6	6	15	15					
21	21	7	7	20	20					
24	24	8	8	22	22					
36	36	13	13	23	23					
38	38	16	16	25	25					
41	41	17	17	29	29					
		18	18	31	31					
		19	19	37	37					
		26	26	39	39					
		27	27	46	46					
		28	28							
		30	30							
		32	32							
		33	33							
		34	34							
		35	35							
		40	40							
		42	42							
		43	43							
		44	44							
		45	45							
		47	47							
		48	48							
		49	49							
		50	50							

Table 16: Community comparison of actual and estimated communities with NMI of 1

-

Communities												
Actual	Esti 1	Actual	Esti 2	Actual	Esti 3	Actual	Esti 4	Esti 4 Actual				
1		2		3		4		5				
1	1	2	2	3	3	4	4	10	10			
12	12	5	5	7	7	15	15	11	11			
14	14	6	6	8	8	20	20	23	23			
21	21	9	9	16	16	22	22	33	33			
24	24	13	13	17	17	25	25	37	37			
31	31	19	19	18	18	39	39					
36	36	26			26	46	46					
38	38	27	27	28	28							
41	41	29	29	32	32							
		30	30	34	34							
		35	35	40	40							
		45	45	42	42							
		47	47	43	43							
				44	44							
				48	48							
				49	49							
				50	50							

Table 17: Community comparison of actual and estimated communities with NMI of 0.95

5 Experimental setup, results and evaluation

This chapter discusses the setup of the hard- and software, the data gathering approach, the results and the evaluation of the results.

5.1 Hardware and software setup

There are ten hardware nodes, existing of a Raspberry Pi 3 model B combined with a LoRa/GPS hat (Dragino LoRa/GPS hat for RPi v1.4), as seen in Figure 26. While the BLE part of the simulation is based on Nordic NRF52 devices, Raspberry Pi's were used for testing because they are easier to program. The onboard BLE module of the Raspberry Pi is used for BLE scanning, the LoRa/GPS hat for LoRa and GPS. At the software side, the modules *Noble* and *Bleno* are used for BLE. Noble is a BLE central module, used to scan for other BLE devices. Bleno is a BLE peripheral module that is used for BLE broadcasting. Testing the range at the location shown in Figure 8 gives a range of approximately 70 meters for the build-in BLE module.



Figure 26: The Raspberry Pi model 3B combined with the LoRa/GPS hat

Bleno

Bleno is used to broadcast the *nodename*, corresponding to the nodenumber of the node, and a *serviceUuid*, a unique number that is used to identify which service is broadcast. For example, a health monitor device may broadcast the heartrate using UUID 0x180D and accelerometer information using UUID 0x2A37. In this case, the uuid is used to make sure that the test nodes can be identified when scanning for BLE devices and no other BLE devices are detected. Algorithm 8 shows the corresponding code.

Noble

taining the minute of the current time and the node number, to a text file and adds the nodenumber to an array. If a found node is already present in the array, the code does nothing. Once a minute, the array is cleared and formerly detected nodes are detectable again. Algorithm 9 shows the corresponding code. Figure 27 shows what a file with scan results looks like. In this case the used node is node 10. The first number is the minute of the hour, the second number is the node number. Node 10 detected node 2 at the 32nd minute until the 40th minute of the hour. Node 8 was detected from the 34th untill the 40th minute. Node 4 was detected from the 39th untill the 40th minute.

bleno.startAdvertising(name, serviceUuid) Algorithm 8: BLE advertising using Bleno

```
Result: Scanning (using Noble)
```

```
var noble = require('noble');
```

```
var fs = require('fs');
```

```
var name = nodenumber;
```

```
var allowDuplicates = true;
```

```
var filename = date.getDays + '-' + date.getHours + '-' + date.getMinutes +'.txt';
```

```
var foundNodes = [];
```

```
noble.startScanning(serviceUuid, allowDuplicates);
```

var nodenum = peripheral.advertisement.localName;

```
if (foundnodes.includes(nodenum)) then
```

```
var nodeinfo = " + date.getMinutes + ',' + peripheral.advertisement.localName + ','
```

```
+ peripheral.advertisement.RSSI;
```

```
fs.appendFile(filename, nodeinfo);
```

```
foundnodes.push(nodenum);
```

end

```
if date.getSeconds == 42 then
```

foundNodes = [];

end

Algorithm 9: BLE scanning using Noble

GPS

The nodes are programmed to save the GPS location once a minute and save this to a file, as seen in algorithm 10.



Figure 27: File with BLE scan results. The first number represents the minute of the hour, the second number represents the detected node id and the third number is the signal strength

```
Result: Save GPS location

import GPS;

import datetime;

import time;

now = datetime.datetime.now();

filename = 'GPS' + '-' + now.day + '-' + now.hour + '-' + now.minute + '.txt';

if (now.second == 42) then

| lat = gps.lat;

lon = gps.lon;

f = open(filename,"a+") f.write(now.minute + ',' + lat + ',' + lon + ");

end

Algorithm 10: Save GPS location once a minute
```

5.2 Data gathering approach

Eight scenario's are defined for the gathered dataset, with the different animal social network indicators taken into account. The data is gathered by placing the hardware nodes according to the scenario, let them scan for the other devices and turn them off again. After this, the data is stored in an adjacency matrix for evaluation purposes. This is done for all eight scenario's. The scenario's are shown in Figures 28 and 29. The goal is to see if the actual network corresponds to the estimated network.

Scenario 1 is used to simulate different components in a network, with one isolated node with the goal to see if the components will stay disconnected. Scenario 2 contains two components with one node connecting two communities, used for betweenness, component and community detection. In scenario 3 all nodes are connected, with the idea to see what happens if a lot of devices are broadcasting near each other. The aim of scenario 4 is to







Figure 28: Scenario's 1 – 4

measure betweenness, component and community detection and isolated node detection.

Scenario 5 is another test to see if two components stay separated from each other. In scenario 6, one isolated node is used to see if it will stay isolated from the rest of the group. In scenario 7, the goal is to see if the two components and the isolated nodes will stay isolated. The purpose of scenario 8 is to see if nodes will accidentally get linked together. All scenario's are also checked for their degree.

5.3 Experimental data gathering

Data was gathered by asking participants to pick up a node and directing them in forming the desired network. When using scenario 6 (Figure 29b as an example, nine nodes should be connected and one node should be isolated. The person with the isolated node is thus directed to walk away from the other nodes and stand still at approximately the critical distance. In an earlier test it turned out that this distance is approximately between 70 and 80 meters. Every person stood still holding their nodes until the nodes performed at least six scans, so at least six timeframes per scenario were collected. The reason for this is





Figure 29: Scenario's 5 – 8

that one timeframe is not enough, since the broadcast of a node at the critical distance may sometimes reach a node and sometimes will not reach a node. For this reason, using only one timeframe will not provide enough information and is not sufficient to draw a conclusion from.

Tests were performed at the same location as in Figures 8a and 8b. At the time of testing, multiple (groups of) people were walking in the area. Some people were walking with their dogs.

5.4 Experimental test results and evaluation

As stated in section 5.2, data is collected using ten Raspberry pi's combined with a LoRa/GPS hat. However, it turned out that the Bluetooth module of one Raspberry Pi was defect and as a result it did not record any data and it was unable to broadcast its node id. Due to this reason, the graphs inferred from the data contain only nine nodes instead of ten.

The networks are constructed by checking which broadcasts the nodes have received while maintaining the network topologies as described in section 5.2. The highest received signal strength (RSSI) is -36, the lowest is -107. The data shows that the RSSI of nearby nodes varies between -36 and -75. The RSSi of nodes that are placed at approximately the critical distance varies between -90 and -107. This means that sometimes node a is able to receive the broadcast of node b, but not always. If during six timeframes a node has received at least four broadcasts of another node, an edge is created between the two nodes.

Figure 30 shows the result of the test corresponding to scenario 1. The network (network 1) contains one falsely identified edge (between nodes 3 and 9). In this scenario, node 2 received the broadcast of node 9 twice (with a RSS of -102 and -100). Node 6 received two broadcasts of node 8 (RSSI -101 and -100) and two broadcasts of node 2 (RSSI -101 and -100). Node 9 received four broadcasts of node 3 (RSSI of -102, -101, -98 and -98), thus an unexpected edge is created. Node 9 also received two broadcasts frmom node 8 (RSSI of -100 and -98).

Network 2 (Figure 31) contains two false identified edges (between 6 and 8 and 6 and 2). Node 1 received one broadcast from 6 (-101), node 2 from node 8 (-100) and node 9 (-103). Node 6 received four broadcasts from node 8 (-96, -96, -101, -100) and four broadcasts from node 2 (-102, -100, -99. -102).

In network 3 (Figure 32) all node were close together so there were no falsely identified edges and the network is the same as in the network in Figure 28c.

Network 4 is the same as that in Figure 28d (except for the missing node), but nodes did receive broadcasts with low signal strength from other nodes. Node 2 received two broadcasts from node 7 (-101, -98), two from node 9 (-101, -104). Node 3 received a broadcast from node 4 (-103), node 4 from node 2 (-101). Node 7 received two broadcasts from node 2 (-102 and -107). Node 9 received one broadcast from node 2 (-99).

Network 5 (Figure 34) contains two false identified edges, between nodes 2 and 9 and between nodes 4 and 6. Node 1 received two broadcasts from node 4 (-99, -101) and node 5 (-101). Node 2 received five broadcasts from node 9 (-98, -99, -100, -100, -99) and from node 6 (-101). Node 6 received four broadcasts from node 4 (-103, -103, -98, -98). Node 8 received two broadcasts from node 9 (-102, -104). Node 9 received two broadcasts from node 8 (-102, -105).

Network 6 (Figure 35 is the same as the network in Figure 29b). But node 1 (the isolated node) received one broadcast from node 4 (-103) and node 3 (-100). Nodes 2 and 4 received a broadcast from node 1 (with RSSI of -101 for node 2, -103 for node 4).

Network 7 (Figure 36) contains one falsely identified edge (between nodes 2 and 4). Node 1 received one broadcast from node 6 (-97). Node 2 received four broadcasts from node 4 (-98, -100, -100, -98) and node 4 received four broadcasts from node 2 (-98, -102, -101, -104). Node 3 received a broadcast from node 9 (-104) and node 2 (-101). Node 6 and 7 both received two broadcasts from node 2 (-99 and -100 for node 6, -100 and -102 for node 7).

Network 8 (Figure 37) contains two false identified edges, between nodes 4 and 8 and nodes 1 and 9. Node 4 received four broadcasts from node 8 (-102, -99, -100, -101) and node 8 received four broadcasts from node 4 (-95, -98, -101, -99). Node 9 received five broadcasts from node 1 (-99, -95, -100, -102, -101).



(a) Actual network for scenario 1

(b) Estimated network for scenario 1

Figure 30: The actual and estimated network for scenario 1





(b) Estimated network for scenario 2





(a) Actual network for scenario 3

(b) Estimated network for scenario 3

Figure 32: The actual and estimated network for scenario 3



(a) Actual network for scenario 4

(b) Estimated network for scenario 4

Figure 33: The actual and estimated network for scenario 4



(a) Actual network for scenario 5

(b) Estimated network for scenario 5





(a) Actual network for scenario 6

(b) Estimated network for scenario 6

Figure 35: The actual and estimated network for scenario 6



(a) Actual network for scenario 7

(b) Estimated network for scenario 7



Actual network for scenario 8											Estim	ated ne	etwork f	or scen	ario 8	
	† *	•,	•2	•,	•:	•6	•>	••	T	,	*	•2	•2	•;	•6	•
	•,									,						

(a) Actual network for scenario 8 (b) Estimated network for scenario 8

Figure 37: The actual and estimated network for scenario 8

5.4.1 Isomorphishm

Three graphs were isomorph (scenario 3, 4 and 6), indicating that in those scenarios both graphs were exactly the same, as can be seen in Figures 32, 33, 35. In one of those scenarios all nodes were connected by each other (scenario 3) and in another scenario one node was isolated and the other nodes were connected by each other. Since those are simple structures, the chance that errors occur is low. This is probably the reason why the graphs are exactly the same.

5.4.2 Isolated node detection

Figure 38 shows the offset of the found number of isolated nodes found in the actual network and the estimated network. There is no offset in 75% of the timeframes and an offset of +1 occurs in 25%.

Figure 39 shows the same node score for the isolated nodes, where timeframes without isolated nodes are omitted. In 60% of the timeframes the isolated nodes are the same. The other bars represent 20%.

Most isolated nodes were the same with a small offset of +1. This +1 offset is due to the errors in scenario 7 and 8 (Figures 36 and 37). This also explains the same node score for the isolated nodes with a score of 0.86 for scenario 8 and a score 0.75 for scenario 7.

5.4.3 Betweenness

Figure 40 shows the offset of nodes with high betweenness, compared to the actual graph based on the actual and estimated graph. As seen, in 50% of the cases the offset is 0, in 37.5% the offset is -1 and in 12.5% the offset is -2.

For the node similarity (seen in Figure 16), in 25% of the timeframes the nodes with high betweenness are the same in both graphs. The percentage for the case where no nodes



Figure 38: Experimental results, isolated nodes: Offset of the number of isolated nodes in the actual graph compared to the estimated graph

are the same and the timeframes where the same node score is between 0.5 and 0.59 is 37.5%.

This is probably due to errors in scenario 2 (Figure 31) and scenario 5 (Figure 34) where two components were accidentally linked together. Due to this, the nodes with the false edge score a higher betweenness in the estimated graph than they should have.



Figure 39: Experimental results, isolated nodes: Same node score



Figure 40: Experimental results, betweenness: Offset of the number of nodes with high betweenness in the actual graph compared to the estimated graph



Figure 41: Experimental results, betweenness: Same node score

5.4.4 Node degree

Figure 42 shows the degree offset. In 75% of the cases the amount of nodes with a high degree are the same. An offset of +2 and -3 occurs in 12.5% of the cases. The same node score (Figure 43) shows that most nodes score between 0.5 and 0.59 or between 0.1 and 0.19 (37.5% in both cases).

The offset and similarity for the degree can also be explained by false edges, leading to nodes in the estimated graph with a higher degree than in the actual graph. Due to this, the possibility exists that they are falsely identified as a node with high degree. For instance in scenario 2 (Figure 31, where node 6 has a false link with nodes 2 and 8.

5.4.5 Density

The average density in the actual graphs is 0.4167 with an MAD of 0.2431, the average density in the estimated graphs is 0.4444 with an MAD of 0.2361.

Figure 44 shows the density difference between the actual density of a timeframe and the estimated density of the timeframe. As seen, in most timeframes the difference is between 0 and -0.009 (37.5%), followed by -0.05 and -0.059 (25%). In 12.5% of the cases the score



Figure 42: Experimental results, degree: Offset of the number of nodes with high degree in the actual graph compared to the estimated graph

is between -0.029 and -0.02.

The difference in density is higher than in the simulation. This is probably due to the size of the network. The test networks are small compared to the simulation networks. Due to this, a false edge between nodes has a bigger impact on the density.


Figure 43: Experimental results, degree: Same node score



Figure 44: Experimental results, density: Difference between the actual density and the estimated density

5.4.6 Component detection

Figure 45 shows that in 62.5% of the cases the amount of components is the same. An offset of +1 occurs in 37.5% of the cases. This offset can be explained due to the errors in scenario 1, 5 and 8 (Figures 30, 34 and 37), which all have one component less in the estimated graph than in the actual graph.



Figure 45: Experimental results, component detection: Offset of the number of components in the actual graph compared to the estimated graph

5.4.7 Communities

The Girvan–Newman algorithm used for community detection is giving problems when used on the collected data. An explanation for this can be that the networks are small and the nodes are similar in terms of node degree and betweenness. For instance, in scenario 3 (Figure 32) every node is connected to every node, resulting in a betweenness that is the same for every edge. Since the algorithm tries to remove the edge with the highest betweenness and every betweenness value is the same, the algorithm fails.

Checking the betweenness confirms this. For scenario 1, the betweenness in the actual network is 0 for all edges. In the estimated network, only one edge has a score greater than 0, which is the falsely identified edge. After removing this edge, the actual and estimated network become identical.

For scenario 2, again the two falsely identified edges score a high betweenness in the estimated network. In the actual network, all betweenness scores are 0. The same situation occurred in scenario 5 and 7. After removing the falsely identified links, the actual and estimated network became the same.

5 EXPERIMENTAL SETUP, RESULTS AND EVALUATION

For scenario 8, one edge has a betweenness score greater than 0 in the actual network and two edges have a betweenness score in the estimated network. In this case the algorithm will remove one edge in the actual network and two edges in the estimated network, resulting in nine isolated nodes in both networks. In scenario 3, 4 and 6 every edge in the actual and estimated network has a betweenness score of 0.

6 Conclusion

This chapter contains the conclusion, limitations of the research and future work. The goal is to answer the main question as stated in chapter 1.

6.1 Conclusion

The overall conclusion is that BLE performs a little bit better than LoRa, except for the graph density and component detection, where LoRa performs better. Both technologies are unsuitable for node degree. Making exact measurements when using BLE or LoRa is not possible. This can be explained by the fact that, due to path loss and shadowing phenomena, the range of a transmitted signal may vary, resulting in the situation that nodes are detectable while they are expected to be out of range or vice versa. This results in errors in the estimated graph and this is the reason why those graphs are a mere approximation of the real situation. This explains why in the simulation there is no timeframe where the actual network and the estimated network are **isomorph**, since one wrongly labelled edge means that there is no isomorphism.

The reason why there are isomorphic networks in the gathered dataset is because it are small networks and two of the three networks are networks where (almost) every node is connected with each other, in a controlled environment. This also explains why the experimental results do not show a lot of similarity when compared to the simulation results. The experimental results are a mere confirmation that, due to path loss and shadowing phenomena, errors can occur, leading to a different network structure and thus different network statistics.

For **isolated nodes**, the amount of nodes with a high betweenness is quite good to determine, taking into account that there is a chance that the there is an offset of 1 isolated node. Especially BLE scores good, with no offset in 68% of the cases, against 40% for LoRa. If there are isolated nodes in the network, it is very hard to determine which exact nodes are isolated.

For **betweenness**, the amount of nodes with a high betweenness is quite good to determine. If there is an offset, this offset will most likely be approximately +1 or -1. Again, it is very hard to find out which exact nodes have a high betweenness, only the amount of nodes. In the case of node **degree**, it is not possible to find out how many nodes with a high degree exist in the graph. It is also impossible to find out which exact nodes have a high degree. Both BLE and LoRa are not suitable for inferring the node degree. On the other hand, graph **density** can be very accurately calculated. While there is a difference between the actual density and the estimated density, the difference is very small.

It is possible to infer information about **component detection** from the graphs based on BLE and LoRa when taking into account that the amount of components in the estimated graphs are probably, for BLE, 1 or 2 higher than in reality and for LoRa the offset is 0 or 1. So,

6 CONCLUSION

an estimation of the amount of components can be made. LoRa has the best performance. **Community** detection is very good for both BLE and LoRa, with BLE performing better than LoRa. Exact information about which nodes belong to which community is hard to infer, but the estimations have a good accuracy.

Both BLE and LoRa can be used for inferring animal social network information, but only as a mere estimation. To see what is possible, the examples in Figure 4 are used. Example **A** uses density, which is very good to predict using both BLE and LoRa. Thus, both techniques can be used in this case.

Example **B** uses social structure and components. The social structure can be inferred using BLE and LoRa to the extent that it is possible to identify how many animals are important in the network and not which exact animals. Detecting the exact amount of components is hard because there will probably always be an offset. Taking this into account, it is possible to estimate the amount of components.

Examples **C** and **D** use density, which is already discussed (in example **A**), just as component detection (in example **B**). Example **E** aims to identify which individuals have a high betweenness, to know which animals should be vaccinated. BLE and LoRa are both suitable for inferring how many animals have a high betweenness, but are both unsuitable for identifying which animals have a high betweenness. Both BLE and LoRa are unsuitable for this purpose. The same goes for example **F**, which also aims to identify nodes with a high betweenness.

One of the authors of [2] was contacted and asked for feedback on the thesis, in order to find out how useful the findings in this thesis are in practice. *Snijders* responded with feedback (discussed in section 6.3) and was able to come up with examples of cases where BLE and LoRa can be used to gather animal social network information, of which two of them are discussed below. She only came up with the corresponding papers and some small hints, the text is written by myself.

Cases where BLE and LoRa can be used for collecting information about animal social networks are research to the social behaviour of hyenas and vultures. *llany et al.* [60] studied the social networks of spotted hyenas, using 20 years of observational data. The authors write "Hyenas were assigned to a single observation session if they were found together, separated from other hyenas by at least 200 m, but usually at least 1 km". For this, BLE can be used to find out if hyenas are together and if there are other hyenas within 200 meters. LoRa can be used to find out how many hyenas are further away than 200 meters but are within the range of LoRa. Findings of the authors were that density was lower in wetter years, negatively influencing the social dynamics. They also found that hyenas form more ties with kin than with non-kin and hyenas tend to associate with central individuals. Degree is used in the last two cases, for which BLE and LoRa are unsuitable.

In the other case, *Corts-Avizanda et al.* [61] used a simulation model to test three hypotheses about information transfer between Griffon vultures, which are known to be social

6 CONCLUSION

scavengers, getting their food from carcasses. Carcasses are considered to be hard to find and it is known that vultures do not actively advertise social information, but other vultures are able to infer social information from another vulture's flight behaviour. There are still a lot of questions about how vultures spread social information, leading to different hypotheses. The three hypotheses tested by the authors are: (1) the *nonsocial hypothesis*, where vultures only use personal information to find carcasses and finds them on their own, (2) the *chains of vultures hypothesis*, where vultures follow a vulture that is flying towards a carcass or follow vultures that are following other vultures, and (3) the *local enhancement hypothesis*, where vultures only react to other vultures that are flying and suddenly dive vertically to a carcass. The authors state in the paper that vultures are able to spot other vultures that are several kilometers away.

Findings of the authors are that the *chains of vultures hypothesis* and the *local enhancement hypothesis* outperform the *nonsocial hypothesis*, where the *local enhancement hypothesis* performs best. Recent research however has been based on the *chains of vultures hypothesis*. According to the authors, more empirical research is needed. Since vultures are able to detect other vultures from great distances, it might be possible to use LoRa to gather empirical information regarding this subject.

As the final conclusion, BLE and LoRa can only be used to replace GPS and proximity sensors if it is not a problem that the collected data is an approximation. For example, if the question is, *how many animals should be vaccinated*, instead of *which animals should be vaccinated*. Approximately the amount of animals can be determined, but not which animals. Another example is taken from [2], which is about white-faced capuchin monkeys. When infants from this species have mothers that are highly social and have a high betweennes, the infants are more likely to survive during stable periods, but are more likely to get killed during alpha male replacements. In this example, it is possible to approximately infer how many infants are likely to survive during the stable periods and how many infants will probably die during the male replacement, but not which exact infants. If this is enough to infer information about the social stability of the group, then BLE and LoRa are capable of replacing GPS and proximity sensors. The only metric where BLE and LoRa are accurate is the network density. Both BLE and LoRa are unsuitable for inferring information about node degree. Table 18 shows an overview of the performance per metric.

6.2 Limitations and future work

Limitations in this work are that this work is not verified in real situations but is mostly theoretical. While the performance of BLE is tested using hardware, it was only tested with nine nodes in eight small and controlled scenarios, without the use of animals. The performance of LoRa using hardware is not validated at all. More real tests using hardware and bigger groups are needed to validate the results in this thesis. Another thing worthwhile to investigate is how BLE and LoRa perform on the metrics using different density levels. If the graph density is lower, less nodes are close together and if the graph density is higher, more nodes

Metric	Performance	Notes
Isomorphism	No isomorphism possible	
Isolated nodes	Suitable, BLE performs best	The offset is probably +1 or -1
Betweenness	Suitable, unable to predict ex-	The offset (of amount of nodes)
	act nodes, only the amount of	is probably +1 or -1
	nodes, both BLE and LoRa per-	
	form good	
Degree	Unsuitable	
Density	Very good performance	
Component detec-	There is a high probability of off-	The offset is probably +1 or +2
tion	set, which should be taken into	for BLE and 0 and +1 for LoRa
	account. The exact amount is	
	hard to determine. LoRa per-	
	forms best	
Community	The community structure is	
	good to predict but exact infor-	
	mation is hard to infer. BLE per-	
	forms best	

are together. This might influence the performance of certain metrics. Maybe, for instance, component detection performs better in more dense networks.

Other future work can be to find out if it is possible to use a received signal strength threshold to only detect nodes within a certain distance. For example, only detect nodes with a received signal strength corresponding to approximately 20 meters and find out how accurate this is. Another interesting approach is to find out what happens if multiple time-frames are used to construct one graph. This might increase the accuracy, because if, for example, a graph is constructed from ten timeframes and the broadcast of one node is received only in two of the ten timeframes by other nodes, this node can be filtered out of the graph because it can be assumed that it was an error due to the path loss and shadowing. On the other hand, this might also lead to the loss of information. Another reason why the broadcast of said node has been received in only two out of ten timeframes might be because an animal got (too) close to another (group of) animal(s), got scared and ran away. This can be crucial information that can get lost because it gets labelled as an error.

On a hardware level, it is interesting to see if Bluetooth Mesh can be used in animal social network applications. Bluetooth Mesh is developed for many-to-many device communication in sensor networks in scalable low-power mesh networks. The developers promise low power, a long range and reliable communication with a large number of devices. Using this for animal social network analysis might be interesting. Bluetooth Mesh is relatively new (released in July 2017, approximately one year ago at the time of writing this thesis). While

there has already been research to Bluetooth mesh networks, to my knowledge there has not been research to using Bluetooth Mesh animal social networking applications.

6.3 Feedback from a biologist

To be able to include a practical perspective and to find out how usable the findings in this thesis are, one of the authors of [2], *Lysanne Snijders*, has been contacted and asked for feedback on the thesis.

To clarify things, the content of sections 6.1 and 6.2 was written before contacting Lysanne Snijders. The only influence that she had on the conclusion (section 6.1) was the inclusion of the examples regarding Hyenas and Vultures.

She wrote that the typical distances used to detect social contact are usually 10 to 15 meters and 2 to 5 meters for disease spreading. She also provided examples where longer distances can be used (the examples with Hyenas and vultures), as discussed in section 6.1. More general, the current data gathering approach using BLE and/or LoRa is most suitable for gathering data about nomadic and solitair animals. However, the problem with this is that those animals do not have a lot of social interaction, resulting in the situation that it takes a lot of time (months, maybe years) to gather enough information. In the case of vultures, for example, the amount of interaction depends on the season, drought and the amount of carcasses available.

The approach of using BLE and/or LoRa does seem useful on a population level, especially for documenting changes that occur on a long term. For this, the absolute number of, for example, components or communities is not essential, but it's more important to know if the amount is increasing or decreasing. The approach is not suitable to use on an individual level, since the approach is unable to identify *which exact* animals are important in the network. Biologists want to link the importance of an animal to certain characteristics, like sex, age and personality, giving insight in natural selection on an individual level. This cannot be achieved by gathering social network information using BLE and/or LoRa in the current manner. It would be useful to optimise the approach to detect social interaction on shorter distances.

References

- [1] A. Goldsmith, Wireless communications. Cambridge university press, 2005.
- [2] L. Snijders, D. Blumstein, C. Stanley, and D. Franks, "Animal social network theory can help wildlife conservation," *Trends in Ecology and Evolution*, vol. 32, no. 8, pp. 567 – 577, aug 2017.
- [3] A. Sedghi, "Facebook: 10 years of social networking, in numbers," Feb 2014. [Online]. Available: https://www.theguardian.com/news/datablog/2014/feb/04/facebook-in-numbers-statistics
- [4] J. Constine, "Instagram hits 1 billion monthly users, up from 800m in september," Jun 2018. [Online]. Available: https://techcrunch.com/2018/06/20/ instagram-1-billion-users/
- [5] D. Karamshuk, C. Boldrini, M. Conti, and A. Passarella, "Human mobility models for opportunistic networks," *IEEE Communications Magazine*, vol. 49, no. 12, pp. 157–165, 2011.
- [6] D. R. Farine and H. Whitehead, "Constructing, conducting and interpreting animal social network analysis," *Journal of Animal Ecology*, vol. 84, no. 5, pp. 1144–1163, 2015.
- [7] "Er zijn 25.000 bedreigde diersoorten, welke verliezen we als eerst?" Jun 2018. [Online]. Available: https://nos.nl/op3/artikel/ 2235445-er-zijn-25-000-bedreigde-diersoorten-welke-verliezen-we-als-eerst.html
- [8] "mens veroorzaakt wereldwijde uitstervingsgolf'," Jun 2015. [Online]. Available: https://nos.nl/op3/artikel/2042853-mens-veroorzaakt-wereldwijde-uitstervingsgolf.html
- [9] R. Kays, M. C. Crofoot, W. Jetz, and M. Wikelski, "Terrestrial animal tracking as an eye on life and planet," *Science*, vol. 348, no. 6240, 2015. [Online]. Available: http://science.sciencemag.org/content/348/6240/aaa2478
- [10] C. C. Wilmers, B. Nickel, C. M. Bryce, J. A. Smith, R. E. Wheat, and V. Yovovich, "The golden age of bio-logging: how animal-borne sensors are advancing the frontiers of ecology," *Ecology*, vol. 96, no. 7, pp. 1741–1753, 2015.
- [11] E. D. Ayele, K. Das, N. Meratnia, and P. J. Havinga, "Leveraging ble and lora in iot network for wildlife monitoring system (wms)," in 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT). IEEE, 2016.
- [12] S. P. Borgatti, A. Mehra, D. J. Brass, and G. Labianca, "Network analysis in the social sciences," *science*, vol. 323, no. 5916, pp. 892–895, 2009.
- [13] S. Sintos and P. Tsaparas, "Using strong triadic closure to characterize ties in social networks," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 1466–1475.

- [14] D. Easley and J. Kleinberg, *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge University Press, 2010.
- [15] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annual review of sociology*, vol. 27, no. 1, pp. 415–444, 2001.
- [16] J. Constine, "Facebook now has 2 billion monthly users... and responsibility," Jun 2017. [Online]. Available: https://techcrunch.com/2017/06/27/facebook-2-billion-users/
- [17] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel, "You are who you know: Inferring user profiles in online social networks," in *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, ser. WSDM '10. New York, NY, USA: ACM, 2010, pp. 251–260. [Online]. Available: http://doi.acm.org/10.1145/1718487.1718519
- [18] M. Turcule, "Ethical issues concerning online social networks," *Procedia Social and Behavioral Sciences*, vol. 149, pp. 967 972, 2014, IUMEN 2014 From Theory to Inquiry in Social Sciences, Iasi, Romania, 10-12 April 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877042814050307
- [19] A. Langone, "What to know about facebook's cambridge analytica problem," Mar 2018. [Online]. Available: http://time.com/5205314/facebook-cambridge-analytica-breach/
- [20] "Facebook to send cambridge analytica data-use notices to 87 million users today," Apr 2018. [Online]. Available: https://www.nbcnews.com/tech/social-media/ facebook-send-cambridge-analytica-data-use-notices-monday-n863811
- [21] S. Meredith, "Facebook-cambridge analytica: A timeline of the data hijacking scandal," Apr 2018. [Online]. Available: https://www.cnbc.com/2018/04/10/ facebook-cambridge-analytica-a-timeline-of-the-data-hijacking-scandal.html
- [22] C. Ellis, "Does facebook know when you'll die?" Jun 2018. [Online]. Available: https://www.techradar.com/news/does-facebook-know-when-youll-die
- [23] I. Kahanda and J. Neville, "Using transactional information to predict link strength in online social networks." *ICWSM*, vol. 9, pp. 74–81, 2009.
- [24] S. Liu, Y. Jiang, and A. Striegel, "Face-to-face proximity estimationusing bluetooth on smartphones," *IEEE Transactions on Mobile Computing*, vol. 13, no. 4, pp. 811–823, 2014.
- [25] C. S. Fischer, *To dwell among friends: Personal networks in town and city*. University of chicago Press, 1982.
- [26] J. Stehlé, N. Voirin, A. Barrat, C. Cattuto, V. Colizza, L. Isella, C. Régis, J.-F. Pinton, N. Khanafer, W. Van den Broeck *et al.*, "Simulation of an seir infectious disease model on the dynamic contact network of conference attendees," *BMC medicine*, vol. 9, no. 1, p. 87, 2011.

- [27] L. Isella, J. Stehlé, A. Barrat, C. Cattuto, J.-F. Pinton, and W. Van den Broeck, "What's in a crowd? analysis of face-to-face behavioral networks," *Journal of theoretical biology*, vol. 271, no. 1, pp. 166–180, 2011.
- [28] M. Génois, C. L. Vestergaard, J. Fournet, A. Panisson, I. Bonmarin, and A. Barrat, "Data on face-to-face contacts in an office building suggest a low-cost vaccination strategy based on community linkers," *Network Science*, vol. 3, no. 3, pp. 326–347, 2015.
- [29] Z. Yu, H. Wang, B. Guo, T. Gu, and T. Mei, "Supporting serendipitous social interaction using human mobility prediction," *IEEE Transactions on Human-Machine Systems*, vol. 45, no. 6, pp. 811–818, 2015.
- [30] N. Cheng, P. Mohapatra, M. Cunche, M. A. Kaafar, R. Boreli, and S. Krishnamurthy, "Inferring user relationship from hidden information in wlans," in *MILITARY COMMUNI-CATIONS CONFERENCE, 2012-MILCOM 2012.* IEEE, 2012, pp. 1–6.
- [31] D. Jacoby and R. Freeman, "Emerging network-based tools in movement ecology," *Trends in Ecology and Evolution*, vol. 31, no. 4, pp. 301 – 314, apr 2016.
- [32] R. Kaysa, S. Tilake, B. Kranstauberc, P. A. Jansenb, C. Carboned, M. J. Rowcliffed, T. Fountaine, J. Eggertf, and Z. Hef, "Monitoring wild animal communities with arrays of motion sensitive camera traps," *arXiv preprint arXiv:1009.5718*, 2010.
- [33] D. T. Blumstein, D. J. Mennill, P. Clemins, L. Girod, K. Yao, G. Patricelli, J. L. Deppe, A. H. Krakauer, C. Clark, K. A. Cortopassi *et al.*, "Acoustic monitoring in terrestrial environments using microphone arrays: applications, technological considerations and prospectus," *Journal of Applied Ecology*, vol. 48, no. 3, pp. 758–767, 2011.
- [34] D. J. Mennill, M. Battiston, D. R. Wilson, J. R. Foote, and S. M. Doucet, "Field test of an affordable, portable, wireless microphone array for spatial monitoring of animal ecology and behaviour," *Methods in Ecology and Evolution*, vol. 3, no. 4, pp. 704–712, 2012.
- [35] M. Lettink and D. P. Armstrong, "An introduction to using mark-recapture analysis for monitoring threatened species," *Department of Conservation Technical Series A*, vol. 28, pp. 5–32, 2003.
- [36] S. J. Cooke, "Biotelemetry and biologging in endangered species research and animal conservation: relevance to regional, national, and iucn red list threat assessments," *Endangered species research*, vol. 4, no. 1-2, pp. 165–185, 2008.
- [37] Semtech, "Semtech sx1272 datasheet," 2017.
- [38] Nordic, "nordic nrf52840 datasheet," 2017.
- [39] Dragino, "Lora/gps hat." [Online]. Available: http://wiki.dragino.com/index.php?title= Lora/GPS_HAT
- [40] T. H. Cormen, Introduction to algorithms, 3rd ed. MIT press, 2009.

- [41] K. Ruohonen, "Graph theory; 2008," *Tampere University of Technology*.
- [42] E. S. Minor and D. L. Urban, "A graph-theory framework for evaluating landscape connectivity and conservation planning," *Conservation biology*, vol. 22, no. 2, pp. 297–307, 2008.
- [43] T. F. Coleman and J. J. Moré, "Estimation of sparse jacobian matrices and graph coloring blems," SIAM journal on Numerical Analysis, vol. 20, no. 1, pp. 187–209, 1983.
- [44] K. M. Stehfest, T. A. Patterson, L. Dagorn, K. N. Holland, D. Itano, and J. M. Semmens, "Network analysis of acoustic tracking data reveals the structure and stability of fish aggregations in the ocean," *Animal behaviour*, vol. 85, no. 4, pp. 839–848, 2013.
- [45] S. Fortunato, "Community detection in graphs," *Physics reports*, vol. 486, no. 3-5, pp. 75–174, 2010.
- [46] M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical review E*, vol. 69, no. 2, p. 026113, 2004.
- [47] A. F. McDaid, D. Greene, and N. Hurley, "Normalized mutual information to evaluate overlapping community finding algorithms," arXiv preprint arXiv:1110.2515, 2011.
- [48] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2005, no. 09, p. P09008, 2005.
- [49] C. Beard and W. Stallings, Wireless Communication Networks and Systems. Pearson, 2016.
- [50] M. Woolley, "Bluetooth 5 technical paper."
- [51] B. SIG, "Specification of the bluetooth system," BlueTooth, Tech. Rep., dec 2016.
- [52] BlueTooth, july 2017. [Online]. Available: https://blog.bluetooth.com/ introducing-bluetooth-mesh-networking
- [53] F. Danebjer and C. Schreiter, "Bluetooth mesh interoperability analysis," 2017.
- [54] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, "A study of lora: Long range & low power networks for the internet of things," *Sensors*, vol. 16, no. 9, p. 1466, 2016.
- [55] J. Petajajarvi, K. Mikhaylov, A. Roivainen, T. Hanninen, and M. Pettissalo, "On the coverage of lpwans: range evaluation and channel attenuation model for lora technology," in *ITS Telecommunications (ITST), 2015 14th International Conference on*. IEEE, 2015, pp. 55–59.
- [56] K.-H. Ke, Q.-W. Liang, G.-J. Zeng, J.-H. Lin, and H.-C. Lee, "A lora wireless mesh networking module for campus-scale monitoring: demo abstract," in *Proceedings of the* 16th ACM/IEEE International Conference on Information Processing in Sensor Networks. ACM, 2017, pp. 259–260.

REFERENCES

- [57] L. Alliance, "Lorawan specification."
- [58] Semtech, "Lora modulation basics."
- [59] V. Daiya, J. Ebenezer, S. S. Murty, and B. Raj, "Experimental analysis of rssi for distance and position estimation," in *Recent Trends in Information Technology (ICRTIT), 2011 International Conference on.* IEEE, 2011, pp. 1093–1098.
- [60] A. Ilany, A. S. Booms, and K. E. Holekamp, "Topological effects of network structure on long-term social network dynamics in a wild mammal," *Ecology letters*, vol. 18, no. 7, pp. 687–695, 2015.
- [61] A. Cortés-Avizanda, R. Jovani, J. A. Donázar, and V. Grimm, "Bird sky networks: How do avian scavengers use social information to find carrion?" *Ecology*, vol. 95, no. 7, pp. 1799–1808, 2014.

A Adjacency matrices degree evaluation

This appendix contains the adjacency matrices corresponding to Figures 19a and 19b.

Table	19: Aa	ljacency	matrix fo	r the	actual	' network	corresp	oonding	to	Figure	19a

	1	2	3	4 5	6	7	8 9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
1	1	0	0 1	0 0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	1	0 0	0 1	1	0	0 1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
3	0	0	1 (0 0	0	1	0 0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
4	0	0	0	1 0	0	0	0 0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	1	0 1	D 1	0	0	0 0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
6	0	1	0 1	0 0	1	0	1 0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
7	0	0	1 (0 0	0	1	1 0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0	0	1	1
8	0	0	0	0 0	1	1	1 0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
9	0	1	0 1	0 0	0	0	0 1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
10	0	0	0 1	0 0	0	0	0 0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0 0	0	0	0 1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	1 0	0	0	0 0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0
13	0	1	0	0 1	1	0	0 0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
14	0	0	0	0 0	0	0	0 0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0
15	0	0	0 1	0 0	0	0	0 0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
16	0	0	1 (0 0	1	1	1 0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0
17	0	0	0 1	0 0	0	0	0 0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	
18	0	0	1 (0 0	0	0	0 0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	
19	0	1	0 1	0 0	1	0	1 0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
20	0	0	0	1 0	0	0	0 0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
21	1	0	0 0	0 0	0	0	0 0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	10	0	0	0	
22	0	0	0 0	0 0	0	0	0 0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	
23	0	0	0 1		0	0	0 0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	5	0	0	0	0	0	0	0	0	0	
24	1	0	0 1		0	0	0 0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	5	0	0	0	0	0	0	0	0	0	
25	0	0			1	0		1		0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	-	0			0			0	0	0	
20	0	1					0 0	0		0	1	0	0	1	0		1	0	0	0	0	0	0	1	1	0	0		0	1	0			0	0	0	0	-	0			0			1	0		
21	0	-				0		0		0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1				0				0		
20	0	0	0 1		0	0		0		0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0		1	0	0	0	0	0	0	0	0	0	1					0			0	0	0	
30	0	1	0 0	n 1	1	0		0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0	0	0	<u></u>				0	1		0	0		
31	0	0	0 0		0	0		0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	<u>_</u>	0		0	0		0	0	0	0	
32	0	0	1 1		1	1	1 0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0	0	0	0	0	1	0	0		0	0	0	1	0
33	0	0	0 0	0 0	0	0	0 0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
34	0	0	1 1	0 0	1	1	0 0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0		0	0	0	1	0
35	0	0	1 0	0 0	1	0	0 0	0	0	0	1	0	0	1	0	1	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0		0	0	0	0	0
36	0	0	0 1	0 0	0	0	0 0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0
37	0	0	0 1	0 0	0	0	0 0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
38	1	0	0 1	0 0	0	0	0 0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
39	0	0	0 1	0 0	0	0	0 0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
40	0	0	0 1	0 0	0	1	0 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0	1	0
41	0	0	0 1	0 0	0	0	0 0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
42	0	0	0 0	0 0	0	1	1 0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0	1	0
43	0	0	0 1	0 0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0
44	0	0	0 0	0 0	0	1	0 0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	1	1	1
45	0	0	0 1	0 0	1	0	0 0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
46	0	0	0 1	0 0	0	0	0 0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
47	0	1	0 1	0 1	0	0	0 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
48	0	0	0 1	0 0	0	0	0 0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1
49	0	0	1 (0 0	0	1	0 0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0	0	1	1
50	0	0	1	0 0	0	1	0 0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1

A ADJACENCY MATRICES DEGREE EVALUATION

	1	2 3	4	5	6 7	7 8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37 3	3 39	40	41	42	43	44	45	46	47	48	49	50
1	1	0 0	0 0	0	0 0	0 0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0 0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	1 0	0 0	0	0 0	0 0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0 0	0	0	0	0	0	0	0	0	1	0	0	0
3	0	0 1	0	0	0 .	1 0	0	0	0	0	1	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	0	0 0	0	0	0	0	0	1	0	0	0	0	1	1
4	0	0 0) 1	0	0 0	0 0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1 0	1	0	0	0	0	0	0	0	0	0	0	0
5	0		0 0	1	0 0	o o	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0 0	0	0	0	0	0	0	0	0	1	0	0	0
6	0	0 0	0 0	0	1 (0 0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0 0	0	0	0	0	0	0	1	0	0	0	1	0
7	0	0 1	0	0	0 .	1 1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	1	1	0	0 0	0	1	0	1	1	1	0	0	0	0	1	0
8	0	0 0	0 0	0	0	1 1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0 0	0	0	0	1	0	0	0	0	0	0	1	0
9	0	1 0	0 0	0	0 0	0 0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0 0	0	0	0	0	0	0	0	0	1	0	0	0
10	0	0 0	0 0	0	0 0	0 0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1 0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0 0	0 0	0	0 0	0 0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1 0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0 0	0 0	0	0 0	0 0	0	0	0	1	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0 1	0	0	1	0	0	0	0	0	0	0	0	0
13	0	0 1	0	0	1 0	0 0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	1	1	0	0	1	0	1	0	0	1	0	0 0	0	0	0	0	0	0	1	0	1	0	0	0
14	0		0 0	0	0 0	0 0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0 1	0	0	1	0	0	0	0	0	0	0	0	0
15	0		0 0	0	0 0	0 0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0 0	1	0	0	0	0	0	0	1	0	0	0	0
16	0	0 1	0	0	0 .	1 1	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	0	0	0 0	0	1	0	1	0	0	0	0	0	0	1	1
17	0		0 0	0	0 0	0 0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0 0	0	0	0	0	0	0	0	0	0	1	0	1
18	0	0 1	0	0	0 0	0 0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	1	0	0 0	0	0	0	0	0	0	0	0	0	0	0	1
19	0	1 0	0 0	0	1 0	D 1	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	1	1	0	0	1	0	1	0	1	1	0	0 0	0	0	0	0	0	0	1	0	0	0	0	0
20	0	0 0) 1	0	0 0	0 0	0	0	0	1	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	0	0
21	1	0 0	0 0	0	0 0	0 0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0 1	0	0	1	0	0	0	0	0	0	0	0	0
22	0	0 0) 1	0	0 0	0 0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0 0	1	0	0	0	0	0	0	1	0	0	0	0
23	0	0 0	0 0	0	0 0	0 0	0	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1 0	0	0	0	0	0	0	0	0	0	0	0	0
24	1	0 0	0 0	0	0 0	0 0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0 1	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0 0	0 0	0	0 0	0 0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0 0	0	0	0	0	0	0	0	1	0	0	0	0
26	0	0 1	0	0	1	1 0	0	0	0	0	1	0	0	1	0	1	1	0	0	0	0	0	0	1	0	1	0	0	0	1	0	1	1	0	0 0	0	0	0	1	0	0	1	0	0	0	1	1
27	0	1 0	0 0	1	0 0	0 0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0 0	0 0	0	0	1 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0 0	0	1	0	1	1	1	0	0	0	0	0	0
29	0	0 0	0 0	0	0 0	0 C	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0 0	1	0	0	0	0	0	0	1	0	0	0	0
30	0	1 0	0 0	1	1 (0 C	0	0	0	0	1	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	1	0	0 0	0	0	0	0	0	0	1	0	0	0	1	0
31	0	0 0	0 0	0	0 0	0 C	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0 1	0	0	0	0	0	0	0	0	0	0	0	0
32	0	0 1	0	0	1 .	1 1	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0 0	0	0	0	1	0	1	1	0	0	0	1	0
33	0	0 0	0 0	0	0 0	0 0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	0	0
34	0	1 1	0	0	1 1	1 1	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0 0	0	0	0	0	0	0	1	0	0	0	1	1
35	0	0 1	0	1	1	1 0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0 0	0	0	0	0	0	0	1	0	0	0	1	0
36	1	0 0	1	0	0 0	0 0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0 1	0	0	1	0	0	0	0	0	0	0	0	0
37	0	0 0) 1	0	0 0	0 0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1 0	0	0	0	0	0	0	0	0	0	0	0	0
38	0	0 0	0	0	0 0	0 0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0 1	0	0	0	0	0	0	0	0	0	0	0	0
39	0	0 0) 1	0	0 0	0 0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0 0	1	0	0	0	0	0	0	0	0	0	0	0
40	0	0 0	0	0	0	1 0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0 0	0	1	0	1	1	1	0	0	0	0	0	0
41	0	0 0	0	0	0 0	0 0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0 0	0	0	1	0	0	0	0	0	0	0	0	0
42	0		0	0	0	1 1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0 0	0	1	0	1	1	1	0	0	0	0	1	0
43	0	0 0	0	0	0	1 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0 0	0	1	0	1	1	0	0	0	0	0	0	0
44	0	0 1	0	0	0.	1 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0 0	0	1	0	1	0	1	1	0	0	0	1	1
45	0	0 0	0	0	1 (0 0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0 0	0	0	0	0	0	1	1	0	0	0	0	0
46	0	0 0	0	0	0 0	0 0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0 0	0	0	0	0	0	0	0	1	0	0	0	0
47	0	1 0	0	1	0 0	0 0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	0	0	0	0	0	0	0	0	1	0	0	0
48	0	0 0	0	0	0 0	0 0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	0	0	0	0	0	0	0	0	0	1	1	1
49	0	0 1	0	0	1	1 1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0 0	0	0	0	1	0	1	0	0	0	1	1	1
50	0	n I 1	10	101	010		101	∧	0					1	- 1 I	- 1 I	∩	_	_	∩	0		0	1 I	0			10	10	10	10	1.4	10		0 0	10		10	10	10	1.4	10		0	4	- 1 I	(H L

Table 20: Adjacency matrix for the estimated network corresponding to Figure 19b

B Adjacency matrices component detection

This appendix contains the adjacency matrices corresponding to Figures 22a and 22b.

Table 21: Adjacency matrix corresponding to the graph in Figure 22a

	1	2	3 4	5	6	7	8 9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
1	1	0	0 0	0	0	0	0 0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	1	0 0	1	0	0	0 1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
3	0	0	1 0	0	0	0	0 0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	1
4	0	0	0 1	0	0	0	0 0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	1	0 0	1	0	0	0 1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
6	0	0	0 0	0	1	0	1 0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
7	0	0	0 0	0	0	1	1 0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	0	1	0	1	1	1	0	0	0	0	1	0
8	0	0	0 0	0	1	1	1 0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	1	0 0	1	0	0	0 1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
10	0	0	0 0	0	0	0	0 0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0 0	0	0	0	0 0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0 1	0	0	0	0 0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0
13	0	1	0 0	1	1	0	0 1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
14	1	0	0 0	0	0	0	0 0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
15	0	0	0 1	0	0	0	0 0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
16	0	0	1 0	0	1	1	1 0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0		0	0	0	0	1	0
17	0	0	1 0	0	0	0	0 0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	1
18	0	0	1 0	0	0	0	0 0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
19	0	1	0 0	0	1	0	1 0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
20	0	0	0 1	0	0	0	0 0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
21	1	0	0 0	0	0	0	0 0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0 0	0	0	0	0 0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
23	0	0	0 0	0	0	0	0 0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
24	1	0	0 0	0	0	0	0 0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0 0	0	0	0	0 0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
26	0	0	1 0	0	1	0	0 0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0
27	0	1	0 0	1	0	0	0 1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
28	0	0	0 0	0	0	1	0 0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0
29	0	0	0 0	1	0	0	0 1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
30	0	1	0 0	1	1	0	0 0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	1	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
31	0	0	0 0	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0
32	0	0	1 0	0	1	1	1 0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0
33	0	0	0 0	0	0	0	0 0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
34	0	0	1 0	0	1	1	1 0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0
35	0	0	0 0	0	1	0	0 0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
36	0	0	0 0	0	0	0	0 0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0
37	0	0	0 0	0	0	0	0 0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
38	1	0	0 0	0	0	0	0 0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0		0	0	0	0	0	
39	0	0	0 0	0	0	0	0 0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0		0	1	0	0	0	
40	0	0	1 0	0	10	1	0 0	0	0	0	0	0	0		1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	U	1	0	0	0	U	0	1	0	1	0		0	U	0	1	1	닏
41	0	0		0	10	0	0 0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	U	0	0	0	0	0	1	0	U	0	U	1	0	1	0	U	1	0	0	0	0	U	U	0	0	0
42	0	0	0 0	0	0	1	0 0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	1	0	0	0	0	0	0
43	0	0	0 0	0	0	1	0 0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1		0	0	0	0	0	
44	0	0	1 0	0	10	1	0 0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0		0	0	0	1	1	
45	0	0	UC	0	1	0	0 0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	1	1	0	1	0	0	1	0	0	υ	0	0	0	0	0	0		U	U	0	0	0
46	0	0		0	0	0	0 0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	U	0	0	υ	1	U	0	0	0	0	0	1	U	0	0	0
47	0	1	0 0	1	0	0	0 1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	1	0	0	
48	0	0	0 0	0	0	0	0 0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0		0	0	0	1	0	
49	0	0	1 0	0	0	1	0 0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0	0	\downarrow	0	0	0	0	1	1
50	0	0	1 0	0	0	0	0 0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	1	1

	1	2	3 4	5	6 7	8	9 10	0 1	1 1	2	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
1	1	0	0 0	0	0 0	0	0 0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
2	0	1	0 0	1	1 0	0	1 0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
3	0	0	1 0	0	0 1	0	0 0	0	0	D	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0	0	0	1	0	1	0	0	0	0	0	0	1	1
4	0	0	0 1	0	0 0	0	0 0	0	· ·	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	1	0 0	1	1 0	0	1 0	0	(D	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
6	0	1	0 0	1	1 1	0	0 0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
7	0	0	1 0	0	1 1	1	0 0	0	(0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	0	1	0	1	1	1	0	0	0	0	1	0
8	0	0	0 0	0	0 1	1	0 0	0	(D	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
9	0	1	0 0	1	0 0	0	1 0	0	(0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
10	0	0	0 0	0	0 0	0	0 1	0	(0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0 0	0	0 0	0	0 0	1	(0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0 1	0	0 0	0	0 0	0	· ·	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0
13	0	0	0 0	1	1 0	0	0 0	0	(0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
14	1	0	0 1	0	0 0	0	0 0	0	(0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
15	0	0	0 0	0	0 0	0	0 0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0
16	0	0	0 0	0	1 1	1	0 0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0	0	1	0
17	0	0	1 0	0	0 0	0	0 0	0	(0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
18	0	0	1 0	0	0 1	0	0 0			D	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	1	1	0	0	0	1	1
19	0	1	0 0	0	1 0	1	0 0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0		0	0	0	0	0
20	0	0	0 1	0	0 0	0	0 0	0	1	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
21	0	0	0 0	0	0 0	0	0 0	0	· ·	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0 0	0	0 0	0	0 0	0	(0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
23	0	0	0 0	0	0 0	0	0 0	0	(0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
24	1	0	0 0	0	0 0	0	0 0	0	(D	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
25	0	0	0 0	0	0 0	0	0 0	0	(0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	1 0	0	1 0	0	0 0	0	(0	1	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
27	0	1	0 0	1	0 0	0	1 0	0	(0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	1	0	0	0
28	0	0	0 0	0	0 1	0	0 0	0	(0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0
29	0	0	0 0	1	0 0	0	1 0	0	(0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0
30	0	1	1 0	1	1 0	1	1 0	0	(0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
31	0	0	0 0	0	0 0	0	0 0	0	· ·	1	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
32	0	0	1 0	0	0 1	0	0 0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0
33	0	0	0 0	0	0 0	0	0 1	0	(0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
34	0	0	1 0	0	1 1	1	0 0	0	(0	0	0	0	1	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0
35	0	0	1 0	1	1 0	0	1 0	0	(0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
36	0	0	0 0	0	0 0	0	0 0	0	· ·	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0
37	0	0	0 0	0	0 0	0	0 0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
38	1	0	0 0	0	0 0	0	0 0	0	•	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0
39	0	0	0 0	0	0 0	0	0 0	0	(0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
40	0	0	1 0	0	0 1	1	0 0	0	(0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0
41	0	0	0 0	0	0 0	0	0 0	0	· ·	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1	0	0	0	0	0	0	0	0	0
42	0	0	1 0	0	0 1	1	0 0	0	(0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1	0
43	1	0	0 0	0	0 1	0	0 0	0	(0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0	0	0
44	0	0	0 0	0	0 1	0	0 0	0	(0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0	0	1	1	0
45	0	0	0 0	1	1 0	0	1 0	0	(0	1	0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	1	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
46	0	0	0 0	0	0 0	0	0 0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
47	0	1	0 0	1	0 0	0	1 0	0	(0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
48	0	0	0 0	0	0 0	0	0 0	0	(0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	1
49	0	0	1 0	0	0 1	0	0 0	0	(0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	1	
50	0	0	1 0	0	0 0	0	0 0	0	0	D	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
_																																																

Table 22: Adjacency matrix corresponding to the graph in Figure 22b

C Adjacency matrices community evaluation

This appendix contains the adjacency matrices corresponding to Figures 24 and 25.

Table 23: Adjacency matrix for the actual communities with NMI = 1

	1	2	3	4	5 (6 7	7 8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35 3	36 3	7 38	39	40	41	42	43	44	45	46	47	48	49	50
1	1	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0) 0	1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	1	0	0	1 0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0 0) 0	0	0	0	0	0	0	0	0	0	1	0	0	0
3	0	0	1	0	0 0	1	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0 0) 0	0	0	0	0	0	0	0	0	0	0	0	1	1
4	0	0	0	1	0 0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0 0) 0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	1	0	0	1 1	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1 () ()	0	0	0	0	0	0	0	1	0	1	0	0	0
6	0	0	0	0	1 1	0	1	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1 () ()	0	0	0	0	0	0	0	1	0	0	0	0	0
7	0	0	1	0	0 0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0 0) 0	0	0	0	0	1	1	1	0	0	0	0	1	0
8	0	0	0	0	0 1	1	1	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0 0) ()	0	0	0	0	1	0	0	0	0	0	0	0	0
9	0	0	0	0	0 0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	0 0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0 0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0 0	0 0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0 0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0) 1	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0 0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0 1	0	1	0	0	1	0	0	0	0	0	0	0	0	0
13	0	1	0	0	1 1	0	1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1 (0 0	0	0	0	0	0	0	0	1	0	0	0	0	0
14	0	0	0	0	0 0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0 1	0	1	0	0	1	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0 0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0 0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
16	0	0	1	0	0 1	1	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0 0) 0	0	0	0	0	1	0	0	0	0	0	0	1	0
17	0	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	0	0	0	1	0	0	0	1	0	0	0	1	0	
18	0	0	1	0	0 0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0 0	0 0	0	0	0	0	0	0	0	0	0	0	0	1	0
19	0	1	0	0	1 1	0	1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1 (0 0	0	0	0	0	0	0	0	1	0	0	0	0	0
20	0	0	0	1	0 0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0 0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
21	0	0	0	0	0 0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0 1		1	0	0	1	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0 0		0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	1	0	0	0	0 0		0	0	0	0	0	0	0	0	1			0	
23	0	0	0	1	0 0		0	0			1	0	1	0	0	0	0			0	0	1	1	0	0	0		0	0	1	0		0			1	0	0	0	0		0	0	0		0	0	
24	0	0	0	0	0 0		0	0	0			0	0	1	0	0	0	0	1	0	1	0	0	1	0		0	0	0	1	0	0	0			1	0	0	0	0	0		0	1			0	
25	0	1	0	0	1 1			0	0	0	0	1	0	0	0	0	0	1		0	0	0	0		1	0		0	1	0	0		0	1 0		0	0	0	0	0			1				0	
20	0	0	0	0	0 0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0 0		0	0	0	0	0	0	10	0	0	1	0	0	
28	0	0	0	0	0 0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0 0		0	0	0	0	1	1	10	0	0	0	0	0	0
29	0	0	0	0	0 0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0 0		0	1	0	0	0	0	0	0	1	0	0	0	0
30	0	0	0	0	1 1	0	0	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1 0		0	0	0	0	0	0	0	1	0	0	0	0	0
31	0	0	0	0	0 0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0 0		0	0	0	0	0	0	0	0	1	0	0	0	0
32	0	0	1	0	0 1	1	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0 0) 0	0	0	0	0	1	0	0	0	0	0	0	1	0
33	0	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0 0) 0	0	0	0	0	0	0	0	0	0	1	0	0	0
34	0	0	1	0	0 1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0 0) 0	0	0	0	0	1	0	0	0	0	0	0	1	0
35	0	0	0	0	1 1	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1 () 0	0	0	0	0	0	0	0	1	0	0	0	0	0
36	0	0	0	0	0 0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0 1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
37	0	0	0	0	0 0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0 0) 1	0	0	0	0	0	0	0	0	0	0	0	0	0
38	1	0	0	0	0 0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0 0) ()	1	0	0	0	0	0	0	0	0	0	0	0	0
39	0	0	0	0	0 0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0 0) ()	0	1	0	0	0	0	0	0	1	0	0	0	0
40	0	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0) ()	0	0	1	0	0	0	1	0	0	0	1	1	1
41	0	0	0	0	0 0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0 1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
42	0	0	0	0	0 0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0 0	0 0	0	0	0	0	1	1	1	0	0	0	0	1	0
43	0	0	0	0	0 0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0 0		0	0	0	0	1	1	0	0	0	0	0	0	0
44	0	0	0	0	0 0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0) 0	0	0	1	0	1	0	1	0	0	0	1	1	1
45	0	0	0	0	1 1	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1 () 0	0	0	0	0	0	0	0	1	0	0	0	0	0
46	0	0	0	0	0 0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	1	0	1	0	0	0	0 0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
47	0	1	0	0	1 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0 0) 0	0	0	0	0	0	0	0	0	0	1	0	0	0
48	0	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0) 0	0	0	1	0	0	0	1	0	0	0	1	0	1
49	0	0	1	0	0 0	1	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0 0	0	0	0	1	0	1	0	1	0	0	0	0	1	
50	0	0	1	0	0 0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	0 0	0	0	1	0	0	0	1	0	0	0	1	1	1

Table 24: Adjacency matrix for the estimated communities with NMI = 1

	1	2 .	1	5	6	7	8 0	10	11	12	13	14	15	16	17	18	10	20	21	22	23	24	25	26	27	28	20	30	31	32	33	34	35	36	37 38	30	40	41	42	43	44	45	46	47	48	10	50
-	+	2 .	, 4	0	0	<u>'</u>				12	13	14	15	10	0	10	19	20	21	22	23	24	25	20	21	20	29	30	31	32	33	34	35	30	0 1	39	40	41	42	43	44	40	40	4 /	40	49	0
2	0		0	1	1			0		0	1	0	0	0	0	0	1			0	0	0	0	1	0	0	0	0	0	0	0	0	0	0		0	0	0	0			0					-
2	0		0	1	-			0				0	0	1	1	1				0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0 0	0	1	0	0			0					-
3	0		1	0				0	0	1	0	0	0	1	0	0	0			0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0 0	0		0	0			0	0				-
4	0		1	0	0			0	10		10	0	0	0	0	0	0			0	0	0	1	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	10	10	0		1			-
5	0	1 0	0	1	1	0		0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0 0	0	0	0	0	0	0	1	0	$\frac{1}{2}$	0	0	0
6	0	1 1	0	1	1			0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0 0	0	0	0	0	0	0	1	0		0	0	0
1	0	0 0	0	0	0	1	1 0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0 0	0	0	0	1	1	1	0	0	0	0		0
8	0	1 0	0	1	0	1	1 0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0 0	0	0	0	0	0	0	0	0	0	0	1	0
9	0		0	0	0	0 0	0 1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	0	0
10	0		0	0	0			1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0 0	0	0	0	0	0	0	0	0		0	0	-
11	0		0	0	0	0 1	J 1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1 0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0 0	1	0	0	0 1	0 0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0 1	0	0	1	0	0	0	0	0	0	0	0	0
13	0	1 0	0	1	1	0	1 0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0 0	0	0	0	0	0	0	1	0	1	0	0	0
14	0	0 0	0	0	0	0 0	0 0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0 1	0	0	1	0	0	0	0	0	0	0	0	0
15	0	0 0	0	0	0	0 1	0 0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0 0	1	0	0	0	0	0	0	1	0	0	0	0
16	0	0 1	0	0	1	1 0	0 0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0 0	0	0	0	1	0	0	1	0	0	0		0
17	0	0 1	0	0	0	1 0	0 0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0 0	0	1	0	0	0	0	0	0	0	1	0	1
18	0		0	0	0	0 0		0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0 0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	1 0	0	1	1	0	1 0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	0 0	0	0	0	0	0	0	1		0	0	0	0
20	0		1	0	0			0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0 0	1	0	0	0	0	0	0	1		0	0	0
21	0		0	0	0		0 0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0 1	0	0	1	0	0	0	0	0		0	0	0
22	0		0	0	0	0 1	0 0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0 0	0	0	0	0	0	0	0	1	0	0	0	0
23	0		0	0	0	0 0	0 0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1 0	0	0	0	0	0	0	0	0	0	0	0	0
24	1		0	0	0		0 0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0 1	0	0	0	0	1	0	0	0		0	0	-
25	0	0 0	1	0	0	0 1	0 0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0 0	1	0	0	0	0	0	0	1	0	0	0	0
26	0	1 0	0	1	1	0 1	0 0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	1	0	0 0	0	0	0	0	0	0	1	0	1	0	0	0
27	0		0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0 0	0	0	0	0	1	0	0					0
20	0		0	0	0							0	1	0	0	0				0	0	0	0	0	0	1	1	0	0	0	0	0	0	0		0	0	0				0	1				0
29	0		0	1	1			0	0		1	0	1	0	0	0	1			0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0 0	0	0	0	0			1	-				0
30	0		0	1				1	0	0		0	0	0	0	0				1	1	0	1	1	0	0	0	1	1	1	0	1	-	0	0 0	0	0	0	0				1			0	0
31	0		0	0	1				0	0	0	0	0	1	1	1	0			1	1	0		0	0	0	0	1	0	1	0	0	1	0	0 0	0	1		0		0	0					1
32	0		0	0				0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0 0	0		0	0		0	0		1			-
24	0		0	0	1	1 1						0	0	1	0	0				0	0	0	0	0	0	0	0	1	0	1	0	1	1	0		0	0	0			1	0		t-		<u> </u>	-
35	0		0	1	1	0		0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0		0	0	0	0	0	10	1	0	0			-
36	0		0	0	0			0	0	1	0	0	0	0	0	0		0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0 1	0	0	1	0	0	0	0	10				-
37	0		0	0	0	0 1		0	1	0	0	0	0	0	0	0	0	0		0	1	0	0	0	0	0	0	0	0	0	0	0	0	<u>.</u>	1 0	0	0	0	0	10	0	0	10				
38	1		1	0	0			0		1	0	1	0	0	0	0			1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0 1	0	0	1	0	0	10	0					-
30	0		0	0	0			0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0		1	0	0	0	0	0	0	1	0			-
40	0		0	0	0			0	0	0	0	0	0	0	1	0	0			0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0 0	0	1	0	0		1	0	10			-	-
40	0		0	0	0			0		1		1	0	0		0	0		1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0 0	0	0	1	0	0		0					+
41	0		0	0	0	1 1		0	0	0	0	0	0	1	0	0	0	0		0	0	0	0	0	0	1	0	0	0	0	0	0	0	0		0	0	0	1	1	1	0	10			1	1
43	0		0	0	0	1 1		0	10	10	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0		0	0	<u> </u>	0 0	0	0	0	1		1	0	10	0			-
44	0		0	0		1 4		10	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		1	0	0		0	1	0				0	10	l o	$\left \frac{1}{1} \right $	$\frac{1}{1}$	$\frac{}{1}$
45	0		0	1	1	0 0		0	10	10	1	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0		0	1	<u> </u>	0 0	0	0	0	10	10	10	1	10	0			-
46	0		0	0	0	0 1		0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	1	0	1	0	0	0	0	.	0 0	1	0	0	0	0	0	0	1	0	10	0	-
47	0		0	1	0	0 0		0	10	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0		0	0	0	0 0	0	0	0	0	10	0	0	0	Ŭ,	tõ l	0	0
48	0		0	0		0 0		0	10	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	0	1	0	0	10	1	0	10	0	$\left \frac{1}{1} \right $	0	1
49	0		0	0		1		10		10		1 0	0	1	0	0		0	0	0	0	0	0	0	0	1	0	0	0	1		1	0	0	0 0	0	1	0	1	1 o		10	10	0		1	1
50	0		0	0	1 0	0 0		0		10		0	0	0	1	0		0	0	0	0	0	0	0	0	0	0	0	0	1	l o		0	0		0	1	10	1		1	0	10	0	$\frac{1}{1}$	it.	$\frac{1}{1}$

Table 25: Adjacenc	y matrix for the actual commu	unities with NMI = 0.95

	1	2	3	4	5 6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37 38	3 3	9 40	41	42	43	44	45	46	47	48	49	50
1	1	0	0 0	0 (0 0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0 1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	1	0 0	0 .	1 0	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0 0	0	0	0	0	0	0	0	0	1	0	0	0
3	0	0	1 (0 (0 0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0 0	0	1	0	0	0	0	0	0	0	0	1	1
4	0	0	0	1 (0 0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	1	0 0	0.	1 0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0 0	0	0	0	0	0	0	0	0	1	0	0	0
6	0	0	0 0	0 () 1	0	1	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0 0	0	0	0	0	0	0	1	0	0	0	0	0
7	0	0	0 0	0 (0 0	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0 0	0	1	0	1	0	1	0	0	0	0	1	0
8	0	0	0 0	0 () 1	1	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0 0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	1	0 0	0	1 0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0 0	0	0	0	0	0	0	0	0	1	0	0	0
10	0	0	0 (0 (0 0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0 (0 (0 0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1 0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0 (0 (0 0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0 1	0	0	1	0	0	0	0	0	0	0	0	0
13	0	1	0 (0.	1 1	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	1	0	0 0	0	0	0	0	0	0	1	0	0	0	0	0
14	1	0	0 (0 (0 0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0 1	0	0	0	0	1	0	0	0	0	0	0	0
15	0	0	0 (0 (0 0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	1	0	0	0	0	0	0	1	0	0	0	0
16	0	0	1 (0 () 1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0 0	0	1	0	1	0	1	0	0	0	0	1	0
17	0	0	1 (0 (0 0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	0	1	0	0	0	0	0	0	0	1	1	1
18	0	0	1 (0 (0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	1	1
19	0	1	0 0	0 0) 1	0	1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0 0	0	0	0	0	0	0	1	0	0	0	0	0
20	0	0	0	1 (0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0 0	1	0	0	0	0	0	0	1	0	0	0	0
21	1	0	0 0		0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0 1	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0 0		0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	1	0	0		0
23	0	0	0 0			0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1 0	0	0	0	0	0	0	0	0	0	0		
24		0	0 0			0		0	0	0	0	0	0	0	0	0	0	0	1		1	0	1	1	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	1	0	0		
25	0	0	1 1			0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0		0	0	0	0	0	0	1	0	0	0	1	
20	0	1	0 0			0	0	1	0	0	0	1	0	0	0	0	0	1	0		0	0	0	0	0	1	0	1	1	0	1	0	0	0	0		0	0	0	0	0	0		0	1	0		
27	0	0	0 0			1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0		0	0	0	1	1	0	0	0	0	0	0	
29	0	0	0 0		1 0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0		0	0	0	0	0	0	1	0	1	0	0	Ū,
30	0	0	0 0	0 .	1 1	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	1	0	1	1	0	0	0	0	1	0	0 0	0	0	0	0	0	0	1	0	0	0	0	ō
31	0	0	0 0	0 0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0 1	0	0	0	0	0	0	0	0	0	0	0	0
32	0	0	1 (0 0	$\frac{1}{1}$	1	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	0	0 0	0	1	0	0	0	0	1	0	0	0	1	0
33	0	0	0 0	0 (0 0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	0	0
34	0	0	1 (0 () 1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	0	0 0	0	0	0	1	0	0	0	0	0	0	1	0
35	0	0	0 (0 () 1	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0 0	0	0	0	0	0	0	1	0	0	0	0	0
36	0	0	0 0	0 (0 0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0 1	0	0	1	0	0	0	0	0	0	0	0	0
37	0	0	0 (0 (0 0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1 0	0	0	0	0	0	0	0	0	0	0	0	0
38	1	0	0 (0 (0 0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0 1	0	0	0	0	0	0	0	0	0	0	0	0
39	0	0	0 0	0 (0 0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	1	0	0	0	0	0	0	1	0	0	0	0
40	0	0	1 (0 (0 0	1	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0 0	0	1	0	1	0	1	0	0	0	1	1	1
41	0	0	0 0	0 (0 0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0 0	0	0	1	0	0	0	0	0	0	0	0	0
42	0	0	0 0	0 (0 0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0 0	0	1	0	1	1	1	0	0	0	0	0	0
43	0	0	0 0	0 (0 0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0 0	0	0	0	1	1	0	0	0	0	0	0	0
44	0	0	0 0	0 (0 0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	0	1	0	1	0	1	0	0	0	1	1	1
45	0	0	0 (0 () 1	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	1	0	1	0	0	1	0	0 0	0	0	0	0	0	0	1	0	0	0	0	0
46	0	0	0 (0 (0 0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0 0	1	0	0	0	0	0	0	1	0	0	0	0
47	0	1	0 (0	1 0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0 0	0	0	0	0	0	0	0	0	1	0	0	0
48	0	0	0 (0 (0 0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	0	1	0	0	0	1	0	0	0	1	0	1
49	0	0	1 (0 (0 0	1	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0 0	0	1	0	0	0	1	0	0	0	0	1	1
50	0	0	1 (0 0	0 0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	0	1	0	0	0	1	0	0	0	1	1	1

	1	2	3	4	5	6 7	7 8	3 9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37 3	8	39 4	0	41	42	43	44	45	46	47	48	49	50
1	1	0	0	0	0 0	0 0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0 1		0 0		0	0	0	0	0	0	0	0	0	0
2	0	1	0	0	1 (0 0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	1	0	0 0)	0 0		0	0	0	0	0	0	0	0	0	0
3	0	0	1	0	0 0	0 0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	1	0	0 0)	0 0		0	0	0	1	0	0	0	1	1	0
4	0	0	0	1	0 0	0 0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0		0 0		0	0	0	0	0	1	0	0	0	0
5	0	1	0	0	1 (0 0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0 0)	0 0		0	0	0	0	0	0	1	0	0	0
6	0	0	0	0	0 1	1 0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	1	0	1	0	1	1	0	0 0)	0 0		0	0	0	0	1	0	0	0	0	0
7	0	0	0	0	0 0) 1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0 0)	0 0		0	1	1	1	0	0	0	0	1	0
8	0	0	0	0	0 0) 1	1	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0	0	0 0)	0 0		0	1	0	0	0	0	0	0	0	0
9	0	1	0	0	1 () (0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0 0)	0 0		0	0	0	0	0	0	1	0	0	0
10	0	0	0	0	0 0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1 ()	0 0		0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0 0	0 0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1 ()	0 0		0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0 0	0 0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0 1		0 0		1	0	0	0	0	0	0	0	0	0
13	0	1	0	0	0 1	1 0	1	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0 0)	0 0		0	0	0	0	1	0	1	0	0	0
14	0	0	0	0	0 0	0 0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0 1		0 0		0	0	1	0	0	0	0	0	0	0
15	0	0	0	0	0 0	0 0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0)	1 0		0	0	0	0	0	1	0	0	0	0
16	0	0	1	0	0 0) 1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	1	0	0 0)	0 1		0	1	0	1	1	0	0	0	1	0
17	0	0	0	0	0 0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0)	0 0		0	0	0	0	0	0	0	1	1	1
18	0	0	1	0	0 0		0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	1	0	0 0)	0 1		0	0	0	0	0	0	0	0	1	0
19	0	0	0	0	0 1	1 0	1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	1	0	0 0)	0 0		0	0	0	0	1	0	0	0	0	0
20	0	0	0	1	0 0		0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 ()	0 0		0	0	0	0	0	1	0	0	0	0
21	1	0	0	0	0 0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0 1		0 0		1	0	0	0	0	0	0	0	0	0
22	0	0	0	1	0 0		0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1 ()	1 0		0	0	0	0	0	1	0	0	0	0
23	0	0	0	0	0 0		0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1 ()	0 0		0	0	0	0	0	0	0	0	0	0
24	1	0	0	0	0 0		1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0 1	_	0 0	-	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0 0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0 0)	0 0		0	0	0	0	0	0	0	0	0	0
26	0	0	1	0	0 1			0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0 (2		-		0	0	0	1	0	0	0	1	
27	0	1	0	0	1 (0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0 0)		-		0	0	0	1	0	1	0	0	0
28	0	0	0	0	0 0			0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0 (2		-		1	1	0	0	0	0	0	0	
29	0	1	0	0	1 1			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0 (2		-		0	0	0	1	0		0	0	
30	0	1	0	0	0 1			1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	1	1	0 0	'				0	0	0	-	0	0	0	+-	
31	0	0	1	0	0 0			0	0	0		0	0	0	1	0	1		0	0	0	0	0	0	1	0	0	0	1	1	1	0	0	1	-	0 0					1		0	-	0	-	0	1	
32	0	0	0	0					1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	1	0		0						0			0	0		0	+	
24	0	0	0	0			1	0			0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	1	0	1	1	0						0		-	0	0		0	1	
25	0	1	1	0	0 1				0	0	0	1	0	0	1	0	1	1	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	1	0	0 0					0			1	0		0		
36	0	0	0	0				0	0	0	1		0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	<u>.</u>	1	0 1	'		-	1	0			<u>-</u>	0		0	10	
37	0	0	0	0				0	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1 0	,				0		-	0	0	-	0	10	
38	1	0	0	0	0 0			0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0 1					0	0	-	0	0	-	0	10	
39	0	0	0	0	0 0			0	0	0	0	0	0	1	0	0	0	0	0	0	1	0		0	0	0	0	0	0	0	0	0	0	0	0	0 0		1 0			0	0	<u> </u>	0	0	-	0	0	
40	0	0	0	0	0 0			0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0 0		0 1			1	0	1	0	0	-	1	0	1
41	0	0	0	0	0 0		0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0 0	,	0 0	+	1	0	0	0	0	0	0	0	0	0
42	0	0	0	0	0 0	$\frac{1}{2}$	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0 0		0 1		0	1	1	1	0	0	-	0	0	0
43	0	0	0	0	0 0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0 0	,	0 0		0	1	1	0	0	0	0	0	0	0
44	0	0	1	0	0 0) 1	10	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0 0	,	0 1		0	1	0	$\frac{1}{1}$	0	0	0	1	1	$\frac{1}{1}$
45	0	0	0	0	0 1	1 0	0	0	0	10	0	1	0	0	1	0	0	1	0	0	0	0	0	0	1	1	0	1	1	0	0	0	0	1	0	0 0	,	0 0			0	0	0	1	0	0	0	0	\mathbf{t}
46	0	0	0	1	0 0		10	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	,	0 0		0	0	0	Ó	0	1	0	0	0	1
47	0	0	0	0	1 0		0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0 0)	0 0		0	0	0	0	0	0	1	0	0	10
48	0	0	1	0	0 0		0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0	,	0 1		0	0	0	1	0	0	0	1	1	1
49	0	0	1	0	0 0) 1	10	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	0	0	0 0	,	0 0		0	0	0	1	0	0	0	1	1	1
50	0	0	0	0	0 0	5 0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0 0)	0 1		0	0	0	1	0	0	0	1	1	1
_	-		_			_	_								_																								_										

Table 26: Adjacency matrix for the estimated communities with NMI = 0.95