



Simulation of social media networks with agent-based modeling

The research and evaluation of using Netlogo, an agent-based modeling software

Frida Gullichsen

Degree thesis

Mediekultur / Online Media

2019

EXAMENSARBETE	
Arcada	
Utbildning:	Mediekultur / Online Media
Identifikationsnummer:	6785
Författare:	Frida Gullichsen
Arbetets namn:	Simulating social networks with agent-based modeling. Research and evaluation of Netlogo, an agent-based modeling software.
Handledare (Arcada):	Owen Kelly
<p>Sammandrag:</p> <p>I dagens samhälle sker många av våra sociala interaktioner på nätet och i stora mängder, då vi kan umgås eller interagera med hundratals, om inte tusentals "vänner" under korta tidsperioder. Vi kan fråga oss om detta är det bästa sättet för oss att interagera med varandra. Evolutionära psykologer hävdar att människor inte kan upprätthålla mer än 150 sociala interaktioner samtidigt. Detta för att det speglar den för oss naturliga sociala struktur som vi människor har. Mitt motiv för detta examensarbete var att hitta ett verktyg som kunde visa hur sociala förbindelser bildas och förändras. Agent-baserad modellering är en datorsimulering där programmerings-objekt eller så kallade agenter, kodas in i en miljö och när modellen körs kan man observera hur agenterna interagerar med varandra och deras miljö, samt observera de mönster som uppkommer under simuleringen. Agent-baserade modeller används för simulering av allt från hur virus sprider sig, hur människor tar genvägar i parker till ekonomiska modeller, men framför allt samhällsvetenskap. Samhällsvetenskap med agent-baserade modeller kan hjälpa simulera och forska i människors beteende. Agent-baserade modeller kan simulera och undersöka system i en simpel och kontrollerad miljö som kan nära motsvara verkliga förhållanden. Experiment i riktiga världen kostar både tid och pengar och agent-baserad modellering ger användaren utrymme att utföra upprepade experiment och såvida insamla värdefull data för sin forskning. Netlogo är det vanligaste agent-baserade modelleringsprogrammet med en lång tröskel för nybörjare att komma igång, samt ett brett utbud av existerande modeller. Modellerna finns inbyggda i programmet men det finns även en mängd av modeller tillgängliga på Netlogos online portal, där användare av Netlogo har laddat upp sina färdiga modeller. Färdiga modeller kan observeras, modifieras eller användas som en grund i byggandet av en ny modell. Detta examensarbete presenterar grunderna till agent-baserad modellering och Netlogo som en agent-baserad modelleringsplattform, samt en nätverksbaserad modell vid namnet Small Worlds, från Netlogos modellbibliotek, för att sedan avgöra om Netlogo är ett användbart forskningsverktyg för användning på Arcada. Examensarbetets slutsats var att Netlogo erbjuder en potentiell lösning för kartläggning av online sociala nätverk. Det kan fungera som ett värdefullt forskningsverktyg i studerandet av social media, även om agent-baserad modellering kräver en viss mängd för att sätta sig in i det.</p>	
Nyckelord:	agent-baserade modeller, ABM, social media, datorsimulation
Sidantal:	26
Språk:	Engelska
Datum för godkännande:	23.5

DEGREE THESIS	
Arcada	
Education:	Mediekultur / Online Media
Identification number:	6785
Author:	Frida Gullichsen
Title:	Simulating social networks with agent-based modeling. Research and evaluation of Netlogo, an agent-based modeling software.
Supervisor (Arcada):	Owen Kelly
<p>Abstract:</p> <p>In today's society, many of our social interactions take place online where we can interact with hundreds, if not thousands, of "friends" in short periods of time. We might find us asking ourselves if this is the best way for us to interact with each other. Evolutionary psychologists claim that humans cannot maintain more than 150 social interactions simultaneously. This, because it reflects the natural social structure that we humans have. My motive for this thesis was to find a tool that could show how social connections are formed and changed. Agent-based modeling is a computer simulation where programming objects or so-called agents are encoded into an environment and when the model is run one can observe how the agents interact with each other and their environment, and observe the patterns that arise during the simulation. Agent-based models are used to simulate everything from how viruses spread, how people take shortcuts in parks to economic models, but especially social science. Social science with agent-based models can help simulate and research human behavior. Agent-based models can simulate and investigate systems in a simple and controlled environment that can closely match real life conditions. Real-world experiments cost both time and money, and agent-based modeling gives the user room to perform repeated experiments and aid in collecting valuable data for their research. Netlogo is the most common agent-based modeling program with a low threshold for beginners to get started, as well as a wide range of existing models to use. The models are built into the program but there are also a number of models available on Netlogo's online portal, where users of Netlogo have uploaded their finished models. Ready models can be observed, modified or used as a base in constructing a new model. This thesis presents the basics of agent-based modeling and introduces Netlogo as an agent-based modeling platform, as well as a network-based model called Small Worlds, from Netlogo's model library, in order to determine whether Netlogo is a useful research tool for use at Arcada. The conclusion of the thesis was that Netlogo offers a potential solution for mapping online social networks. It can serve as a valuable research tool in the study of social media, although agent-based modeling requires some effort to set up.</p>	
Nyckelord:	agent-based models, ABM, social media, computer simulation
Number of pages:	26
Language:	English
Date of acceptance:	23.5

OPINNÄYTE	
Arcada	
Koulutus:	Mediekultur / Online Media
Tunnistenumero:	6785
Tekijä:	Frida Gullichsen
Työn nimi:	Simulating social networks with agent-based modeling. Research and evaluation of Netlogo, an agent-based modeling software.
Työn ohjaaja (Arcada):	Owen Kelly
<p>Tiivistelmä:</p> <p>Maaailma, jossa suuri osa sosiaalisesta interaktiosta tapahtuu verkossa, saa meidät usein miettimään onko se paras muoto kanssakäymiseen. Evolutiiviset psykologit väittävät, etteivät ihmiset voi meidän sosiaalisen evoluution mukaan ylläpitää yli 150 sosiaalista suhdetta, silti verkossa useimmat ihmiset sanovat “ystäviä”, joiden kanssa pidetään yhteyttä, olevan satoja, ellei jopa tuhansia. Motiivini tähän tutkimukseen oli löytää työkalu, jolla voisi tutkia tapaa, jolla sosiaaliset suhteet muodostuvat ja muuttuvat. Agenttipohjainen mallinnus on tietokonesimulaatio jossa ohjelmointikohteita, niin sanotusti agentteja, koodataan ympäristöön ja kuin malli pyöritään, voidaan havaita miten agentit käyttäytyvät yhdessä tai ympäristönsä kanssa, ja tarkkailla simulaation aikana syntyviä kuvioita. Agenttipohjainen mallinnus simuloi monimutkaisia järjestelmiä, kuten metsäpalojen leviäminen tai liikennemuutokset mutta myös erityisesti sosiaalitiede. Sosiaalitiede voi agenttipohjaisen mallinnuksen avulla simuloida ja observoida ihmisen käyttäytymistä. Agenttipohjainen mallinnus voi mallintaa, simuloida ja tutkia vallitsevia yhteiskuntajärjestelmiä kontrolloidussa mutta yksinkertaistetussa ympäristössä, joka tiiviisti arvioi todennäköisyyksiä tutkimuksen alaisuudessa ja simuloivat oikeita elinolosuhteita. Reaalimaailman tutkimukset maksavat sekä rahaa että aikaa, ja agenttipohjainen mallinnus antaa käyttäjälle mahdollisuuden suorittaa toistuvia kokeita ja auttaa keräämään arvokkaita tietoja tutkimukseensa. Netlogo on yleisimmin käytetty agenttipohjainen mallinnustyökalu, jonka käytön aloittaminen on tehty aloittelijoille helpoksi ja josta löytyy laaja kattaus valmiiksi tehtyjä malleja. Malleja voi tutkia, muokata ja käyttää uusien mallien pohjana. Tämä tutkimus käy läpi agenttipohjaisen mallinnuksen perusteet, Netlogo ohjelmana sekä Netlogon mallikirjastosta muodostuvan verkkopohjaisen mallin nimeltään Small Worlds, tavoitteena määrittää mikäli se tarjoaa hyvän tutkimustyökalun Arcadalle. Tutkimus selvitti, että Netlogo tarjoaa potentiaalisen ratkaisun sosiaalisten verkostojen kartoitukseen, ja voi tarjota arvoa tuovan lisätyökalun jokaiseen tutkimukseen, vaikkakin vaatii hieman vaivannäköä asennukseen.</p>	
Avainsanat:	agentti-pohjainen mallitus, ABM, sosiaalinen media, tietokonesimulaatio, simulaatio
Sivumäärä:	26
Kieli:	englanti
Hyväksymispäivämäärä:	23.5

CONTENT

1 INTRODUCTION	6
1.1 Background & motive	7
1.2 Objective	7
1.3 Context	8
1.4 Delimitation	10
1.5 Structure & methods	10
1.6 Core definitions	11
2 THEORY OF ABM	14
2.1 Agent-based modeling (ABM)	14
2.2 Complex Systems and Emergence	15
2.3 Agents	16
2.4 ABM and behavioural research	17
2.5 Social network Analysis (SNA)	17
3 NETLOGO	18
3.1 Netlogo basics	19
3.2 Networks in Netlogo	23
3.3 Using Data	23
3.4 Summary	24
4 CASE STUDY: SMALL-WORLD MODEL	25
4.1 The model	25
4.2 Parametres of the model	27
4.3 Summary	29
5 CONCLUSION	30
REFERENCES	33
APPENDICE BILAGOR	36

1 INTRODUCTION

Online social networks and the way we interact with each other in them has repeatedly been proven to contradict the natural social structure of humans. Humans are said not to be able to maintain more than 150 social connections (Robert Dunbar, 2014). The way we interact with each other on existing online social networks does not mirror a social structure we strive in - they are not compact even though they comprise of hundreds or even thousands of people. One could argue this might lead to a less integrated society.

What can we do to counteract this? How is it that social networks go wrong in the social structure they create? Is there a way for us to observe this or maybe even take the first step to fix it?

Agent-based models, or ABMs, are computer simulations where programming objects called “agents” are coded into an environment and when the model is run, one can observe how the agents interact with each other as well as their environment. It is used for a variety of things, anything from architecture (Chen 2012) to mapping economic models to virus spread . Social sciences is an area agent-based modelling has been increasingly used in, the studying of human behaviour, as to make more sense of the complex nature of us human beings (Klein & Marx & Fischbach 2018).

This thesis will research the usage of agent-based models, specifically a program called Netlogo, an agent-based modeling platform program that you download onto your desktop. It is the most widely used software and Netlogo as a software application could be a tool to help prove that the way we connect on Facebook or similar social media platforms is not a natural way to interact. I will present and analyse my findings through the research done, as well as argue for and against the usage of Netlogo, what one would need to know and think about when constructing a model as well as the challenges of building a model from scratch. This thesis is intended to serve as a basis for further research at Arcada or as a gateway for others to try out what Netlogo has to offer.

1.1 Background & motive

The starting point for this thesis came from a research subject by my mentor Doctor Owen Kelly presented at an earlier time. The research would problematise the current structure of social networks in social media because it is an argument to not fulfil the evolutionary natural structure of us humans (Roberts & Arrow & Gowlett & Lehmann & Dunbar 2014). The hypothesis of the research would itself around that our social structure acts in ripples, “social ripples” or “pools” if you will, with inner and outer circles of friendships. Facebook and other social media do not reflect this. Dr Kelly wanted to research, analyse and on the basis of that possibly implement a new form of social media, that would act as a tool for us to extend our abilities as individuals instead of shaping them to what we choose to project. This became the basis for my thesis. Furthermore, my own motivation for the project was also based in the belief that our current social media platforms are not doing us good and anything to prove that right would be one step closer to fighting it.

1.2 Objective

Social networks have been modeled in Netlogo, but my research involved getting to know the software and determining if it was suitable use by the department. This involved discovering how easy it is for a committed to read and observe Netlogo models. It also involved determining to what extent pre-existing models are readily available and easily modified. The research necessarily required summarising and simplifying the available information and then attempting to implement my own research by testing existing models in Netlogo.

This thesis aims to present:

- What agent-based models are and what they are used for

- Introduce Netlogo as a platform
- Present a network-based Netlogo model
- Draw conclusions about Netlogo's suitability for use in the department

Goals of the thesis:

- Gain an understanding of the uses of Netlogo and ABM in general
- Determine if Netlogo and ABM provide suitable tools for researchers in the Institute of Culture and Media at Arcada

Since I had no prior knowledge of agent-based modeling or Netlogo, this thesis assumes the reader to be in a similar state of knowledge.

1.3 Context

Early on, I discovered that the topic at its very beginning - social networks and what is wrong with them - was extremely broad and needed to be narrowed down right from the start. In order to analyse and observe how connections form in social media networks, or what motivates individuals to interact with each other, observations would need to be viewed and documented. Simply analysing one's own networks would only prove a single case. Was there a way for us to see networks or connections unfold and collect data from this? Could this help us prove something about the connections that could hint at the abnormality of it all?

During my research, I started out exploring the options within the field of agent-based modeling, a computer simulation. What programs are available and what skill levels do they require? Hence ABM being a computer simulation, knowing or being familiar with a programming language became evident at the get-go. Myself, being familiar only with front-end development, decided to research the alternatives for beginners. During my

initial research the softwares I found suitable were Ascape, Netlogo, MASON and Repast.

Many agent-based modeling softwares are designed as university projects to be used in specific research only or as academic projects with niched uses. Many of them also belong to different branches of the same ABM family tree and build upon each other. These four above ABM softwares all feature a good amount of documentation as well as ready models to work with, which is great for a beginner, not being forced to start from scratch. All of them operate in the programming language Java, except for Netlogo (Agent based models 2019). Netlogo being the only of these that has developed its own programming language, carrying the same name as the software itself. The language was developed to be understood by younger students, yet still carrying the level of complexity to perform scientific studies or researches with. Netlogo was also my platform of choice because of it carrying the biggest library of pre-existing models, all of varying complexity and difficulty, giving users of all levels a wide variety of choice to experiment or build upon.

This was not the end of my delimitation. As I started analysing agent-based modeling, I realised there was plenty of software out there, but many of the applications had very steep learning curves. After pouring through user-manuals for programs such as Ascape (Centola 2010), and reading about the connections between the social sciences and agent-based modeling, it was needed to figure out what would be the main focus of the thesis. It wasn't until I came across Netlogo, the most user-friendly of ABM software, that the pieces started falling into place. Netlogo seemed easy to use, both for those coding simulations as well as those observing them. This would be helpful in showcasing something like the complex structure of social network connections.

1.4 Delimitation

I set out to learn the basics of Netlogo, here still aiming at constructing my own model to present the features of the software, as well as showcasing the simplicity around which it revolves. After learning the basics of the software it became evident quite early on that building a model myself it would require research more suitable for a master thesis. Thus, the focus shifted to finding a way of portraying the software within its context, shedding light on how it works and why or why not I find it suitable for the intended use.

1.5 Structure & methods

This thesis is theory heavy, aiming to give the reader an overview of agent-based modeling, Netlogo as a software and the pros and cons of using it. Firstly, I will give an introduction to agent-based modeling, a concept that applies to Netlogo and several other platforms. They all adhere to the same ground rules, some requiring more coding, some requiring more math. There is a plethora of platforms to choose from. I predominantly aim to present the basics of Netlogo as an agent-based modeling platform.

- **Theory**

Present the basics of agent-based modeling, which are the same for all platforms. This will give an overview of the components that structure a model.

- **Netlogo as a platform**

What is Netlogo and how does it work? This chapter aims to give an insight in both the simple and complex pieces of Netlogo as a software

- **Netlogo a case study**

This chapter explores a model more in-depth for the reader to get a basic understanding of Netlogo's possibilities and how one could extend a model.

- **Conclusion**

Drawing a conclusion to my research and why I drew an end to where I did. The conclusion also aims to present where and how to get started if one were to embark upon exploring agent-based modeling and Netlogo.

The material I used mainly consists of online material found related to agent-based modeling. Reading through scientific texts and aiming to simplify them within the concept of mapping networks.

1.6 Core definitions

Agent-based modelling (ABM)

Agent-based modelling (ABM) is a type of computer simulation where actors, so-called “agents” of different kinds and their behaviour and their environment is coded to represent a phenomenon of some kind, like a traffic jam or a virus spread. When the simulation is run, you can observe and change parameters accordingly to make assumptions and conclusions and draw parallels to real life situations and problems.

Agent

An individual, an animal, a group, a cell is represented as a programming object. This agent is assigned different attributes or classes that will define its actions and behaviour during the simulation.

Complex system

ABM are complex systems. Nature is a complex system. An important attribute of a complex system is that no individual component drives the behaviour of the system. The behaviour of the system is the result of diverse interactions between the components within.

Simulation

Simulation in ABM can mean two things. The first application of the word simulation is a noun that refers to the entire process of using simulation techniques to solve a specific problem. These techniques include analysis of the problem, the development of a simulation model, the simulation itself, as well as the analysis of the result. The second interpretation is the verb used for the act of simulation with the help of a simulation program.

Social Network Analysis (SNA)

Social network analysis (SNA) is a process where social structures are examined through the use of networks and graph theory.

Model

A model is an idealised simplified representation of reality. The purpose of the model is to enable studies of specific situations in society, for example, which are easier to observe in a visual simulation than in the real world.

Netlogo

A platform that uses its own agent-based modeling language in a relatively simple simulation environment, where models can be constructed to simulate complex natural and social systems.

Ascape

An agent-based modeling framework that uses the programming language Java.

Mason

A multiagent simulation library core. It is designed to work as the foundation for larger simulations in the programming language Java.

Repast

An open source agent-based modeling toolkit. Models are built with the programming language Java. The toolkit is free and simplifies both model creation and use.

2 THEORY OF ABM

2.1 Agent-based modeling (ABM)

Agent-based modeling, or ABM, is a powerful technique in simulating and exploring real-world phenomena through computational simulation. A real-world phenomenon meaning anything from how an epidemic spreads or how traffic jams form, to the small world theory to wealth distribution. Essentially, almost any complex system can be reconstructed in agent-based modeling (Wilensky & Rand 2015). In ABM, a system is modelled as a set of autonomous agents that the designer gives attributes to and then observe the agents make decisions and interact with each other and their environment (Coursera 2016).

So what does an agent-based model consist of? The model starts off with a concept of a model, a real-world complex system that you want modeled (Wilensky & Rand 2015). This is a big chunk of the research work. Defining what you want to observe is one thing, but seeing it as a complex system, such as an ecosystem or a social structure in which everything is connected to everything else, identifying the other relevant parts to observe (and code) can be challenging, because of all the components to model.

Once you have your complex system mapped out, the actual model consists of three components:

- **Agents**
- **Environment**
- **Time**

You have your **agents**, the individuals, or entities observed within the model. Agents are given attributes within a model like a goal, to communicate, to move or just to exist

within the model. The **environment** they are in can serve as a neutral base or an environment that affects or influences the behaviour of the agents. The role of the environment will, of course, affect the outcome of the model.

Time is the third component of your model. When each model or simulation is run, it goes through a number of time steps or ticks. These metrics can represent a second, a day or a year. Depending on the duration of the tick, different behaviour can be implemented between the agents or in the environment.

Lastly, the rules that tie it all together. The rules define under which conditions all interactions take place and evolve; how the agents interact with each other and interact with their environment. Are the rules fixed or do they change according to how the model evolves? Do they define behaviour or do they define action?

ABM is a visualisation of interactions over time. These interactions are between a number of agents and also their environment. The rules for how often these interactions occur, and how they change the status of the agents involved in any interaction, can be simple or complex. The purpose of an ABM visualisation is to reveal the behaviour of agents by observing the patterns they make. (Macal & North 2016)

2.2 Complex Systems and Emergence

A complex system is a system where every single action that is made affects the system as a whole, like the melting of our polar ice affecting the ocean streams, or the spread of a viral video over the Internet in mere days. For this, a complex system is also one we seek to understand, us humans being on the continuous quest for knowledge and understanding. Computer simulations help us model, simulate and fully investigate such systems in a controlled but simplified environment that can nevertheless closely resemble reality.

In a complex system, where elements are tied together, an “emergent phenomena” becomes apparent. This phenomenon is a key characteristic of complex systems. Emergence can be defined as “the arising of novel and coherent structures, patterns and properties through the interactions of multiple distributed elements”. In a complex system, order can emerge without any design or designer. How can order arise by chance without the help of a designer? But systems are often found to be self-organised and evolved rather than intelligently designed by a designer (Wilensky & Rand 2015).

For example, the first Netlogo model ever was that of a traffic jam. One might assume that the cause of a traffic jam would be the fault of a single car. What emerged from the model was the fact that one car slowing down, even slightly, while entering the highway, caused a ripple effect through the line of cars. This caused all cars to back up until they were able to accelerate again when space freed up in front of them.

In complex systems, you see patterns or structures emerge that you had maybe not expected. In agent-based modeling, when behaviour encoded at the agent level results in unexpected behaviour at a macro level, this is what we call emergence.

2.3 Agents

What is an agent? An agent is a software entity that has properties and follows rules. The person coding the model defines the properties or attributes, and the rules. Agents are a diverse set of individuals and have their own objectives, and can, on the basis of their defined behaviour, adapt and even learn to change their behaviour accordingly.

Furthermore, rules are applied for the interactions between agents, simple or adaptive ones (like the grass is green if you eat it, but if it's brown, you don't).

Within the system set up for them, they can show simple behaviour that indicates they adapt and learn in a situation.

2.4 ABM and behavioural research

ABM offers the behavioural sciences a computational toolkit for developing precise and specific models of how individuals interact, and for discovering patterns of behaviour and organisation that emerge from these interactions. Instead of relying on verbal, qualitative theories, you can build ABMs of the phenomena you are interested in understanding and explaining. The models can then be tested against data and to the degree they successfully agree with the data, we can achieve a deeper and more precise understanding of phenomena we wish to understand, explain, and predict.

2.5 Social network Analysis (SNA)

The study of social networks stems from a desire to explain, understand and model social phenomena in a better way. This can be done by using mathematics, graph theory or statistics. Social structures are a web of relationships that surround actors within a network, actors being people, organisations etc (Scott 1987). Social networks can imply friendships among students in a school, advice giving Facebook communities or even alliances between organisations. Studying the impact of social networks can help us understand and change the way we interact with each other to promote balance.

For anyone wanting to understand social contagion or how social interaction can affect your health, network analysis is the ideal alternative (El-Sayed, Abdulrahman M & Scarborough, Peter & Seemann, Lars & Galea, Sandro (2012). Some social networks can be critical, stigmatising and even aggressive. These do not reflect a healthy relationship between actors.

3 NETLOGO

Netlogo is an agent-based modeling language as well as a programmable environment for simulating real-world phenomena. It first appeared back in 1990, designed by Uri Wilensky and continues to be developed to this day

Many of the programming languages used to build agent-based models can be hard and complicated and require engineer-level knowledge to develop even the simplest of models. Netlogo, on the other hand, is a tool designed to be easy for both designing models as well as observing models. Still, as aforementioned, models do not require a lot of code to show profoundly complex interactions, so Netlogo is also a tool for even the most advanced user as it scales up so easily. Its advantage lies in its low threshold for beginners while not having a steep learning curve. Virtually anybody can learn to code in Netlogo. For this reason, it is also the most widely used agent-based modeling software.

Netlogo was designed to be easily readable but also visually appealing that even a kindergarten student could grasp the most basics concepts. It is based on the programming language Logo, which was created in 1967 by a team including Seymour Papert. Its goal was to create a mathematical simulation where children could play with words and numbers. The most broadly used implementation of Logo was incorporated into the Apple 2 and early Atari and Commodore computers (Wikipedia 2019).

Netlogo is also used by a large number of scientists and other professionals in cutting edge research. This gives Netlogo both a low and a high ceiling, making it ideal for most research settings, be it in a classroom or a research facility.

Furthermore, Netlogo has extensive documentation and tutorials. Inside Netlogo itself lies the Models Library, a large collection of ready models to use and modify. When learning the software, you begin by learning from ready-made simulations. These come in a very wide variety of options, everything from biological phenomena like ants spreading pheromones when foraging for food to geographical events such as the spread of forest fires, to phenomena such as segregation in social science (Wilensky & Rand 2015).

The online platform of Netlogo features a community of users who have uploaded their own models, which are free to download and interact with. This can facilitate working with new or existing models, since other models can provide valuable working code to add to the model you are working with.

3.1 Netlogo basics

Just as in the basics of agent-based modeling, in Netlogo you set up agents by giving them attributes. You set up an environment in which the agents interact, and then make rules under which all these apply. You then run the simulation and observe. The basics are the same for all agent-based modeling environments. The main advantage of Netlogo is its simple interface (Figure 1). The interface is a 2D spatial view of the model environment. All the elements you create in it you can click and drag around the interface, one by one or several at a time. A right-click gives you the option to edit the element as well, creating exactly right interface for both you and others who interact with. The size of the actual grid, the black square where the simulation takes place, can also be changed.

The first tab, *Interface* is where the simulation happens. Every agent and function coded along with the buttons will appear in this view. The grid is made up of cells and all events that appear within it are placed and move around an x and a y axis. If you want to place an agent or code a specific area in the grid, called a patch, coordinates are used. The second tab, *Info*, is where you find information of the code written for the model. The observer can peruse for themselves the basics behind the model, and find out which slight or larger modifications can be made to the model in order to change the outcome of the simulation. The third tab, *Code*, is where all the back-end coding of the model is done.

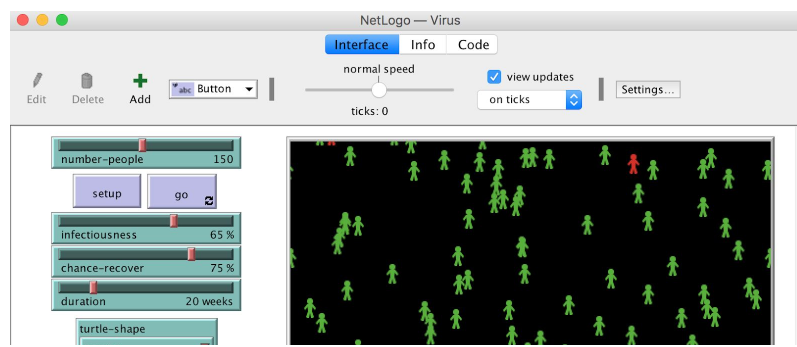


Figure 1. Netlogo Virus model setup (Screen grab from Netlogo)

In the interface, the black window you see represents the world. The world is a grid whose basic regions are called patches. Each patch is a 2x2 grid cell and each patch is identified by the coordinate at its centre. Each patch is a type of agent that can be assigned different attributes and rules under which they interact with other elements in the simulation.

The world can be inhabited by three types of agents: mobile agents, stationary agents and connecting agents. Mobile agents are agents that can move around the simulation - turtles. Stationary agents are patches which cannot move at all but can still contribute to the simulation. Thirdly, the connecting agents are the links and connect two or more agents.

So a turtle is a movable entity within the world and they always have a position and a heading. Other attributes you give them include shape, size and colour. When collecting data, attributes like age or sex can be added to the turtles, but these do not always need to affect the function of the turtles or how they react with others.

The first two buttons you need to get the world running is a setup button, which sets up the world you have created and a run button, that runs the simulation. This all you need to observe a simulation. In the code tab, the setup button could, for instance, be told to create 5 turtles (Figure 2) . When pressed, 5 turtles appear in the simulation. You also tell them to turn clockwise 180 degrees and take one step to the left. When the simulation is set up, and the run button is pressed, the turtles start moving around the simulation until the run button is pressed again and the simulation starts.

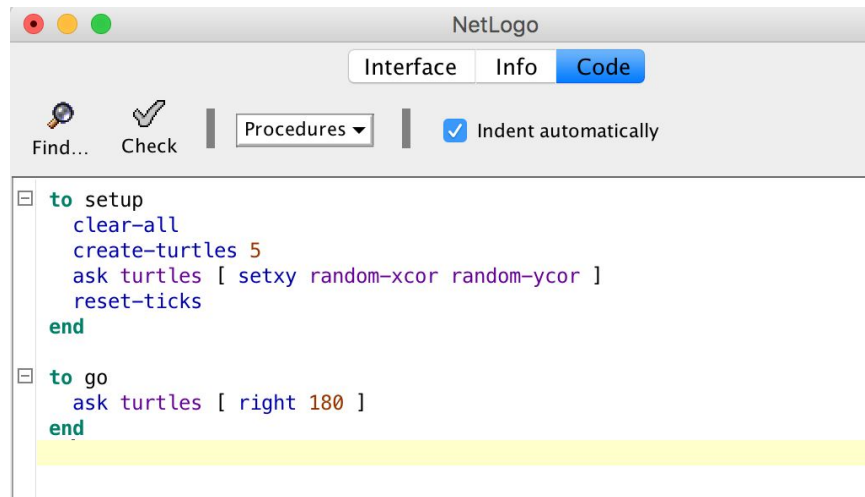


Figure 2. Netlogo code tab (Screen grab from Netlogo)

When you open Netlogo, the world is empty i.e. there is no turtles and the world is black. Turtles need to be created and you set up the world in the code tab in (Figure 1), but turtles can also be created in the Command center by the **Observer**. The observer has statements or commands, that it gives to turtles and patches. Similar to the Console, found after right-clicking to inspect a webpage. Nothing is permanently coded, but commands will be executed immediately, without the need to visit the Code tab.

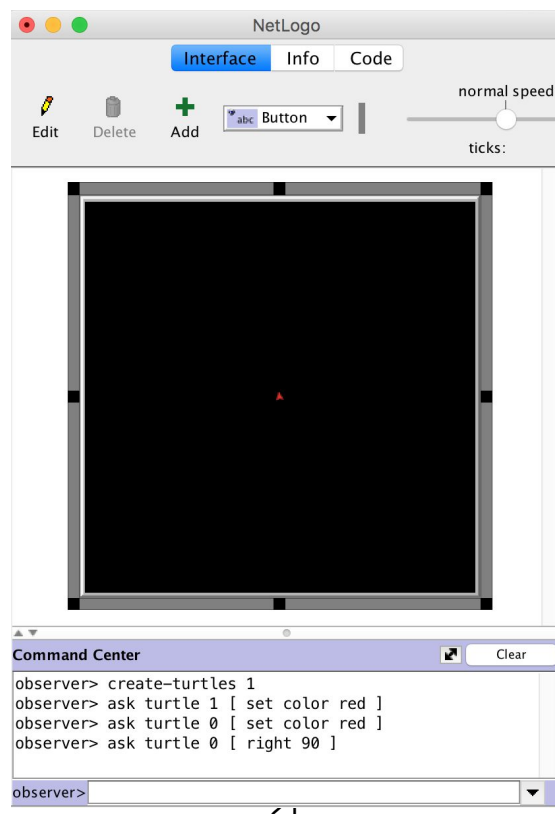


Figure 3. Command center in Netlogo (Screen grab from Netlogo)

```
create-turtles 1: creates one turtle in the centre of the world (first being turtle 0)
inspect turtle 0: shows the properties of turtle 0
```

With the ask command you can specify commands to be run by turtles or patches:

```
ask turtle [ set color red ]: sets the colour of turtles to red
ask turtle 0 [ right 90 ] : makes the turtle turn 90 degrees
```

3.2 Networks in Netlogo

In Netlogo you can set different sceneries with the help of extensions. Networks is an extension for Netlogo, bundled into the software. Network models require not only agents, but agentsets, since networks are made up of nodes interconnecting. Agentsets are named to facilitate calling on them, like bankers and clients (Wilensky 1999 s.303). In the code you define how turtles initially link to each other, and how they reform connections when running the model. For instance, some links connect to nodes with the most connections (popularity), while other links are random (Wilensky 1999). When the simulation is run, you can observe clustering around the nodes with the most connections, basically showing how some networks form. In all social networks, also those online, there are clustering that might be tied to location, work, school or age. In a simulation, these factors could be added to observe further regrouping and clustering.

3.3 Using Data

When working with any research project, getting data to validate your argument is imperative and any computational model amasses a large amount of data. This is because

of the number of inputs and outputs that change in the model, they all produce their own set of information (Wilensky & Rand 2015). In order to get insight into the research conducted, the data needs to be read and analysed. There are a few ways to do this.

- **Raw data**

Making use of raw data is running your model x amount of times and then doing comparative runs. This is tedious and since the models runs on randomness, it might not produce data that can be used on a larger scale.

- **Statistical**

Statistics are the most common way of making use of data. Means, standard deviations, median and more.

- **Graphs**

Simple to produce and easy to understand. Seeing the model evolve in the graph can also be beneficial when comparing results.

- **Spatial**

Addresses analysis of patterns of variables in one two or higher dimensional space.

3.4 Summary

Netlogo is a agent-based modeling software with an easy-to-learn language making it a low-threshold platform to observe and modify existing models as well as building your own. The software features a wide selection of ready-made models in-program as well as downloadable models from the community. All ready-made models are provided with information about the code and modification suggestions, facilitating working and experimenting with existing models without having to build one from scratch.

Networks are created in Netlogo by defining nodes and what they connect with. Currently there is a handful of network models in the Model Library in Netlogo but quite a few more in the community. This would facilitate any project wanting to work with and observe network formation. Data can also be extracted and visually observed to conclude or add to arguments in research settings.

4 CASE STUDY: SMALL-WORLD MODEL

In this chapter, I will do a walk-through of a model in Netlogo. The reason for doing this is first and foremost to give the reader an example of the software in use as well as showcasing a model that exhibits a complex system I first set out to discover - networks.

4.1 The model

The model is called Small Worlds, a network-based model illustrating the theory that any person is only a couple of connections away from any other person in the world. Small-world properties are found in many real-world phenomena, for example, transportation networks, neuron networks, electric power grids and social networks.

The small world's hypothesis originates from several experiments carried out by Stanley Milgram and other researchers in the 1960s. They were studying the average path lengths of social networks in the US and Milgram devised an experiment where individuals from 4 US cities were chosen to randomly receive an information packet which detailed the purpose of the study and basic information about a contact person in Boston who was the target. If the recipient knew this person on a first-name basis, he or she would forward the letter to that person. If they didn't they needed to think of a friend or relative that did and forward the packet to them. This study examined the types of chains they created and

found human society is a type of small world network characterized by short path-lengths. (Travers & Stanley 1969). This experiment as well as the other carried out are often associated with the phrase “six degrees of separation”.

The reason for me choosing this model as a case study is, first, to show a model illustrating social networks similar to something we find on social media. Some share a multitude of friends or acquaintances, others only have a single friend in common and many circles of friends overlap each other. Small worlds within Facebook communities or subreddits on Reddit could also be used comparatively. Second, the model might be fairly complex, but it is an excellent example of showing how anything spreads in a social network - anything from rumours, viral videos or even diseases.

The model is an adaption of a graph model by Duncan Watts and Steve Strogatz called “Collective dynamics of ‘small-world’ networks” (1998). This paper practically launched the science of complex networks with not a lot of material existing about complex and self-organising systems before this. Both their model and this Small-World model are a base for network study with Netlogo, complex networks being either regular, small-world or random (Complex Networks, Small-world, scale-free and beyond).

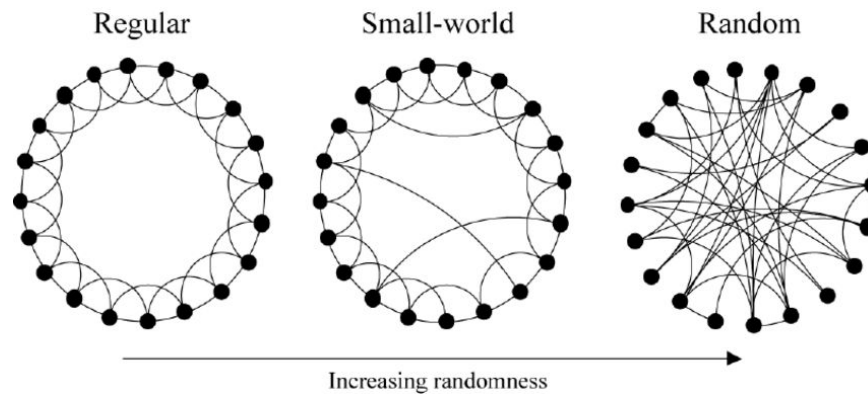


Figure 4. The different types of networks (Hamille & Nigel 2010)

In the small-world network the average degree of separation is small, with actors having both close and distant connections. When talking about networks analysis as well as

network modelling in ABM, you often talk about path lengths and clustering. Real-world networks have short path lengths and high clustering, which neither regular nor random networks have.

The model is a circle network that rewires the endpoint of each link to a random node with a certain probability. This means each node starts out connected to each neighbour and as you rewire them, either one by one or all simultaneously, they randomly connect with a new node within the network, keeping the other end the same.

4.2 Parametres of the model

Getting started you SETUP the model, which places an x amount of nodes in a circle in the world that is black starting out. Each node is connected to two others on either side. The slider named NUM-NODES controls the size of the network. The REWIRE-ALL button creates the network by connecting all nodes and rewires each one so each node is connected to another. Rewiring here means changing one end of a pair of connected nodes while the other stays the same. The REWIRING-PROBABILITY slider determines

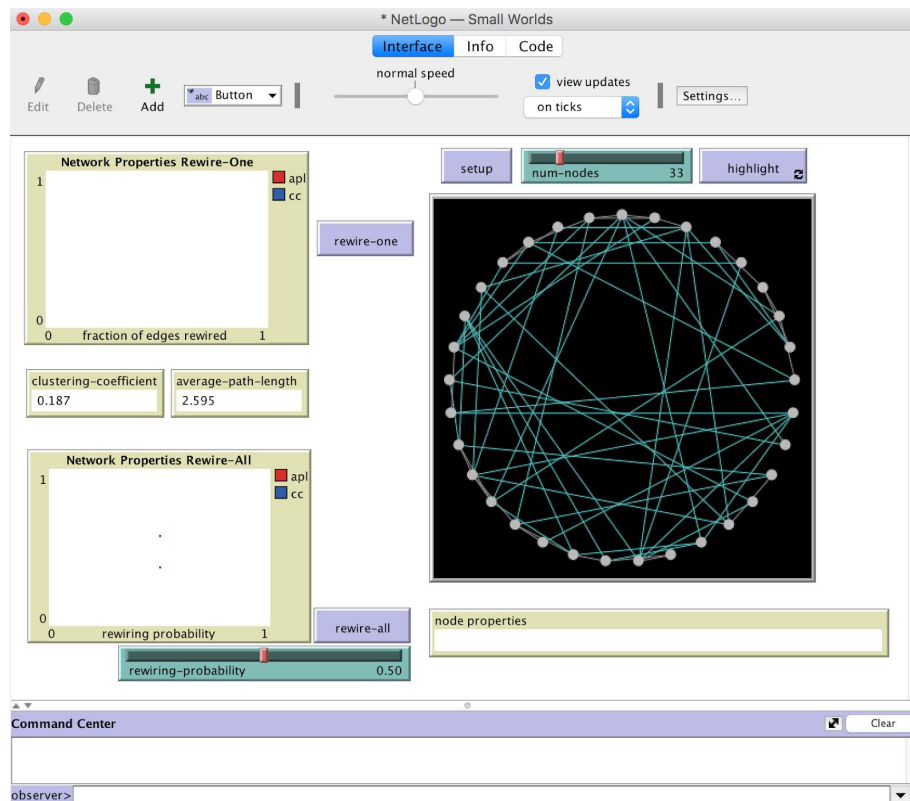


Figure 5. Small Worlds Model (Screen grab from Netlogo)

the probability that an edge will get rewired. Pressing REWIRE-ALL with even a few probabilities produces networks with varying average path lengths and clustering coefficients.

On the left-hand side, you have two monitors showing these two. Path lengths measure the distance between every pair of nodes in the network and then averaging the results. Clustering measures regrouping in local network neighbourhoods and how it deviates from randomness, for example, “What is the probability that two of my friends are also friends?”. The monitors show the values for the whole network. Networks with short average path lengths and high clustering coefficients are considered small world networks.

When you have set up the model you alternate between REWIRE-ALL and REWIRE-ONE. The latter picks only a single random node and rewires it. When you press HIGHLIGHT and hover over a node you can see the node itself turn pink and the nodes it is rewired to blue. The next step of nodes if interconnected are shown with a yellow wire, giving you an indication of the clustering coefficient for the pink node.

The graphs on the side show the how the clustering coefficient and the average path length change when rewiring all or single nodes. As you rewire, you can see the network starting out regular with even close connections between most nodes, then progressing to a small world network with high clustering and short average path lengths, to a random network with no apparent structure (Centola 2017).

These are the basics of the model. Changing the parameters like probability or the set of nodes, lets you observe different results. Furthermore, the model can be extended to for instance follow the spread of a rumour within a network.

4.3 Summary

The Small Worlds model is model illustrating how any person is only a few connections away from anyone else in the world. The model consists of a circle of nodes set to connect with each other. The parameters of the model lets you set the probability of how edges get rewired as well as observing how the model changes rewiring nodes one by one or all simultaneously. The model can also be used to show how quickly something could spread along a network, like a viral video on social media.

5 CONCLUSION

The primary goals of this thesis were to come to an understanding of how Netlogo might be used and to determine whether it might prove a useful research tool for use at Arcada. This involved determining whether Netlogo could be used effectively by researchers with little or no programming experience. To reiterate what was said earlier the choice of Netlogo as the application to test was partly based upon the fact that the programming language Netlogo was designed to be usable by school children of 8 and upwards.

It did not prove possible to approach Netlogo, the application, without some basic knowledge of the programming language. However, since the software has been designed for users of all levels, this basic knowledge should be within the grasp of any interested researcher. This is especially true for anyone wishing to explore models in Netlogo's Model Library.

A book called "An Introduction to Agent-based modeling - Modeling Natural science and engineered complex systems with Netlogo" by Uri Wilensky and William Rand is especially suitable for this stage of interacting with models. Although text heavy this gives an insight into the main uses and an overview of Netlogo's components while taking existing models as examples. Since Uri Wilensky is the creator of Netlogo, as well as a very readable author, the examples are well thought out and thoroughly explained.

For exploring further into the software, researchers would benefit from a basic knowledge of a programming language like Python or Java. While this would be beneficial it is not strictly necessary, although familiarity with coding would remove a large part of the learning curve. Existing tutorials in the aforementioned book as well as in the Netlogo User Manual aid with getting the basics covered. Once the basics are covered, one can start learning to extend basic existing models. The Small World model

can be extended to show how, for example, rumours spread simply by colour coding the nodes and seeing how just after a few rewiring of the nodes results in all nodes so coloured.

One can accomplish a lot of useful research by working with publicly available models and modifying and extending them. Even advanced ABM models can contain part of code borrowed from other models. However, it should be borne in mind that starting out to build a model does not only require knowledge of the programming language but also a clear concept of what one wants to model. Since working with Netlogo follows the same steps as other research work, you need to start out with one or several research questions of a problem you want to solve or phenomena you wish to observe. One also needs to take into account that building a model can be very challenging when the inputs and outputs are numerous.

A simple model is nothing more than a system of agents and the relationships between them, but it can still exhibit a profound amount of complexity and by this provide insight to real-world phenomena by observation in a controlled environment. Experiments in the real-world cost both time and money and, as we all know, real-life environments and individuals are far from controllable. Using ABM and Netlogo will give the observer or the researcher the space to change parameters within the model to take account of a wide range of circumstances. This can create valuable data to develop new hypothesis on the research conducted.

Agent-based modeling has been around for over 50 years, but since the late 1990s it has seen a sharp rise in use in many areas, as mentioned both in biology as well as business and social sciences. One of the reasons is the fact that real-life data from, say, economical models, social network analysis and geographic information systems can be incorporated into agent-based modeling. There are also ways to make ABM's more powerful by using machine learning. Still, as aforementioned, a simple model is many times more than enough to observe a phenomenon. Human beings observe and comprehend their world

through patterns and this is precisely what Netlogo aids with. Building a complex model is far from easy and grasping the concepts behind require a lot of research and work, but models are also built to be observed and interacted with and even modified to yield different results - not everybody has to build models from scratch.

In conclusion it appears that Netlogo offers a potential solution to mapping social networks. It requires some effort to set up, but should justify this wherever researchers have a clear hypothesis they wish to model. However we do it, we need to understand social networks better, and Netlogo offers one method that anyone can use, even without programming experience.

Netlogo can provide a valuable additional tool in any research tool kit designed to explore social media. It also simplifies the equally important process of coming to an understanding of the underlying benefits of agent-based modeling. Any researcher will need to understand what agent-based modeling is and what it can achieve before they can devise a hypothesis to model. Netlogo offers both possibilities.

REFERENCES

Agent based models (2019) [www] Available:
<http://www.agent-based-models.com/blog/resources/simulators/> Referenced: 11.5.2019

Agent based modeling - a simpler way to understand complexity (2016) Coursera [www]
Available:
<https://www.coursera.org/lecture/systems-science-obesity/agent-based-modeling-a-simpler-way-to-understand-complexity-pqYse>

Centola, Damon (2010). *The Ascape Model Developer's Manual* [www] Available:
<http://ascape.sourceforge.net/manual/Overview.html> Referenced: 4.5.2018

Centola, Damon (2017) *2.5 Using Netlogo: Properties of the Small World Network*.
[www]<https://www.coursera.org/lecture/networkdynamics/2-5-using-netlogo-properties-of-the-small-world-network-JOQIF> Referenced: 5.5.2019

Chen, Liang (2012). *Agent-based modeling in urban and architectural research: A brief literature review*. Frontiers of Architectural Research Volume 1, Issue 2, June 2012, Pages 166-177 [www] Available:
<https://www.sciencedirect.com/science/article/pii/S2095263512000167> Referenced: 17.4.2019

El-Sayed, Abdulrahman M & Scarborough, Peter & Seemann, Lars & Galea, Sandro (2012) *Social network analysis and agent-based modeling in social epidemiology*

Hamille, Lynne & Gilbert, Nigel (2010) *Simulating Large Social Networks In Agent-based models: A Social Circle Model*. [www] Available:
http://epubs.surrey.ac.uk/2709/1/Hamill_Gilbert11.pdf Referenced: 17.5.2019

Klein, Dominik & Marx, Johannes & Fischbach, Kai. (2018). *Agent-Based Modeling in Social Science, History, and Philosophy : An Introduction*.

Logo (programming language). (2019) Wikipedia, The Free Encyclopedia [www] Available:

[https://en.wikipedia.org/w/index.php?title=Logo_\(programming_language\)&oldid=897151116](https://en.wikipedia.org/w/index.php?title=Logo_(programming_language)&oldid=897151116)

Macal, Charles & North, Michael (2006) *Tutorial on Agent-based modeling and simulation Part 2: How to Model with Agents* [www] Available: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.450.429&rep=rep1&type=pdf>

Referenced: 14.5.2019

Parker, M (2001). "What is Ascape and Why Should You Care?", *Journal of Artificial Societies and Social Simulation* vol. 4, no 1.

Roberts, Sam & Arrow, Holly & Gowlett, J & Lehmann, Julia & Dunbar, Robin. (2014). *Close Social Relationships*.

Scott, Travis (1987). *Social Network Analysis: a Handbook*

Tisue, Seth & Wilensky, Uri (2004) *Netlogo: A Simple Environment For Modeling Complexity* [www] Available:

https://www.researchgate.net/publication/230818221_NetLogo_A_simple_environment_for_modeling_complexity Referenced: 17.7.2018

Travers, Jeffrey & Milgram, Stanley (1969) *An Experimental Study of the Small World* [www] Available:

<https://snap.stanford.edu/class/cs224w-readings/travers69smallworld.pdf> Referenced: 5.5.2019

Wilensky, Uri. (1999). NetLogo. [www] Available: <http://ccl.northwestern.edu/netlogo/> Referenced: 24.4.2019

Wilensky, Uri (1999). *Netlogo User Manual version 6.0.4* [www] Available: <https://ccl.northwestern.edu/netlogo/docs/> Referenced: 17.7.2018

Wilensky, Uri. & Rand, Michael. (2015), *An introduction to agent-based modeling: modeling natural, social and engineered complex systems with Netlogo*

Wilensky, Uri. (2005). NetLogo Small Worlds model. [www] Available: <http://ccl.northwestern.edu/netlogo/models/SmallWorlds>. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL. Referenced: 24.4.2019

APPENDICE BILAGOR

Appendice 1

Online sociala nätverk och hur vi interagerar med varandra i dem har upprepade gånger visat sig strida mot människans naturliga sociala struktur. Människor sägs inte kunna upprätthålla mer än 150 sociala förbindelser och hur vi interagerar med varandra i dem har upprepade gånger visat sig strida mot den evolutionära sociala strukturen vi som människor har. Människor sägs inte kunna behålla mer än 150 sociala kontakter. Det sätt som vi interagerar med varandra på befintliga sociala nätverk på nätet speglar inte en social struktur vi strävar efter - de är inte kompakta trots att de består av hundratals eller tusentals människor. Man kan hävda att detta kan leda till ett mindre integrerat samhälle. Vad kan vi göra för att motverka detta? Hur är det att sociala nätverk går fel i den sociala strukturen de skapar? Finns det ett sätt för oss att observera detta eller kanske till och med ta det första steget för att fixa det?

Agentbaserade modeller, eller ABM, är datorsimuleringar där programmerings objekt som kallas "agenter" kodas till en miljö och när modellen körs kan man observera hur agenterna interagerar med varandra och deras miljö. Den används för en mängd olika saker, allt från arkitektur till kartläggning av ekonomiska modeller till spridning av virus. Samhällsvetenskap är ett område agentbaserad modellering har i allt högre grad använts i att studera mänskligt beteende för att ge större känsla för människors komplexa natur.

Detta examensarbete kommer att undersöka användningen av agentbaserade modeller, särskilt ett program som heter Netlogo, ett agentbaserat modelleringsprogram som du laddar ner till skrivbordet. Det är den mest använda mjukvaran och Netlogo som ett program kan vara ett verktyg för att bevisa att sättet vi ansluter oss på Facebook eller liknande sociala medieplattformar inte är ett naturligt sätt att interagera. Jag kommer presentera och analysera mina resultat genom den forskning som gjorts, samt argumentera för och mot användningen av Netlogo, vad man behöver veta och tänka på

när man bygger en modell samt utmaningarna att bygga en modell från början. Detta examensarbete är avsedd att ligga till grund för vidare forskning på Arcada eller som en inkörsport för andra att prova vad Netlogo har att erbjuda.

Utgångspunkten för detta examensarbete kom från ett forskningsämne av min mentor doktor Owen Kelly presenterad vid en tidigare tidpunkt. Forskningen skulle problematisera den nuvarande strukturen i sociala nätverk i sociala medier eftersom det är ett argument att inte uppfylla den mänskliga evolutionära naturliga sociala strukturen. Hypotesen av forskningen skulle i sig själva vara att vår sociala struktur verkar i krusningar, "sociala krusningar" eller "pooler" om du vill, med inre och yttre cirklar av vänskap. Facebook och andra sociala medier reflekterar inte detta. Dr Kelly ville undersöka, analysera och på grundval av det eventuellt skapa en ny form av sociala medier som skulle fungera som ett verktyg för oss att utöka våra förmågor som individer istället för att forma dem till det vi väljer att bli. Detta blev grunden för mitt examensarbete. Dessutom var min egen motivation för projektet också baserad på tron att våra nuvarande sociala medier-plattformar inte gör oss gott och allt för att bevisa detta skulle vara ett steg närmare att bekämpa det.

Redan i begynnelse av min research upptäckte jag att ämnet - sociala nätverk och vad som är fel med dem - var extremt brett och måste börja skalas ner. För att kunna analysera och observera hur kontakter bildas i sociala medier, eller vad som motiverar individer att interagera med varandra, måste observationer betraktas och dokumenteras. Att bara analysera sina egna nätverk skulle bara bevisa ett enda enskilt fall. Finns det ett sätt för oss att se nätverk eller kontakter formas och samla in data från detta? Kunde detta hjälpa oss bevisa någonting om förbindelserna som skulle tyda på hur onormalt det är?

Under min forskning började jag utforska alternativen inom agentbaserad modellering. Vilka program är tillgängliga och vilka färdighetsnivåer behöver de? I och med att ABM är en datasimulering, krävdes det att känna till eller vara bekant med ett programmeringsspråk uppenbart från början. Själv är jag bara bekant med front-end

utveckling, så bestämde mig för att undersöka alternativen för nybörjare. Under min första undersökning var de program som jag fann lämpliga Ascape, Netlogo, MASON och Repast.

Många agentbaserade modelleringsprogram är utformade som universitetsprojekt som endast används i specifik forskning eller som akademiska projekt med specifika användningar (Agent-baserade modeller 2019). Många av dem tillhör också olika grenar av samma ABM-släktträd och bygger på varandra. Dessa fyra ovannämnda ABM-programvaror har alla en bra mängd dokumentation och färdiga modeller att arbeta med, vilket är bra för en nybörjare, och tvingar inte användaren att börja helt från början. Alla arbetar i programmeringsspråket Java, förutom Netlogo. Netlogo är den enda av dessa som har utvecklat sitt eget programmeringsspråk, med samma namn som själva mjukvaran. Språket utvecklades för att förstås av yngre studenter, men ändå bär komplexiteten för att utföra vetenskapliga studier eller undersökningar med. Netlogo var också mitt val av plattform på grund av att den bär det största biblioteket av befintliga modeller, alla med varierande komplexitetsnivå och svårighetsgrader, vilket ger användarna på alla nivåer ett brett utbud av modeller att experimentera med eller bygga vidare på.

Detta var inte slutet på min avgränsning. När jag började analysera agentbaserad modellering insåg jag att det fanns gott om programvara där ute, men många av programmen hade väldigt branta inlärningskurvor. Efter att ha gått igenom användarhandböcker för program som Ascape och läst om förbindelserna mellan samhällsvetenskap och agentbaserad modellering, var det nödvändigt att ta reda på vad som skulle vara huvudfokus för examensarbetet. Det var inte förrän jag valde Netlogo, den mest användarvänliga av ABM-programvaran, som bitarna började falla på plats. Netlogo såg lätt ut att använda, både för de som vill programmera modeller samt för de som endast vill observera dem. Detta skulle vara till hjälp vid uppvisning av något som den komplexa strukturen i förbindelser i social nätverk.

Detta examensarbete innehåller mycket teori och syftar till att ge läsaren en överblick över agentbaserad modellering, Netlogo som en mjukvara och för-och nackdelarna med att använda den. För det första kommer jag att ge en introduktion till agentbaserad modellering, grunderna som gäller Netlogo och alla andra ABM plattformar. De följer alla samma regler, vissa kräver mer kodning, vissa kräver mer matematik. Det finns en mängd plattformar att välja mellan. Sedan ämnar jag presentera grunden för Netlogo som en agentbaserad modelleringsplattform.

De primära målen med detta examensarbete skulle vara att förstå hur Netlogo kan användas och för att avgöra om det kan vara ett användbart forskningsverktyg för användning på Arcada. Detta medförde att Netlogo kunde användas effektivt av forskare med liten eller ingen programmeringserfarenhet. För att upprepa vad som sagts tidigare var valet av Netlogo som plattform att testa delvis baserat på det faktum att programmeringsspråket Netlogo var utformat för att användas av skolbarn 8 och uppåt.

Det visade sig inte möjligt att närma sig Netlogo som plattform utan någon grundläggande kunskap om programmeringsspråk. Men eftersom mjukvaran har utformats för användare på alla nivåer, bör denna grundläggande kunskap ligga inom ramen för en intresserad forskare. Detta gäller särskilt för alla som vill utforska modeller i Netlogos Model Library.

En bok som heter "An Introduction to Agent-based modeling - Modeling Natural Science and engineered complex systems with Netlogo" av Uri Wilensky och William Rand är särskilt lämplig för detta stadium av interaktion med modeller. Även om det är tung läsning så ger boken inblick i huvudanvändningen av programmet samt en översikt över Netlogos komponenter med existerande modeller som används till exempel. Eftersom Uri Wilensky är skaparen av Netlogo, och även en mycket läsbar författare, är exemplen väl genomtänkta och grundligt förklarade.

För att utforska sig vidare i mjukvaran skulle potentiella forskare dra nytta av en grundläggande kunskap om ett programmeringsspråk som Python eller Java. Även om detta skulle vara fördelaktigt är det inte absolut nödvändigt, även om kännedom om kodning skulle ta bort en stor del av inlärningskurvan. Befintlig handledning i ovannämnda bok samt i Netlogos användarmanual kan hjälpa med att få grunderna täckta. När grunderna är täckta kan man börja lära sig att utöka grundläggande befintliga modeller. Small World modellen kan utökas för att visa hur exempelvis rykten sprids enkelt genom färgkodning av noderna och att se hur strax efter några omkopplingar av noderna resulterar i alla knutpunkter färgade.

Man kan åstadkomma en hel del användbar forskning genom att arbeta med allmänt tillgängliga modeller och modifiera och utvidga dem. Även avancerade ABM-modeller kan innehålla kod som kan lånas till andra modeller. Man bör emellertid komma ihåg att det inte bara kräver kunskap om programmeringsspråket, men också ett tydligt koncept om vad man vill modellera. Eftersom då du arbetar med Netlogo följer samma steg som andra forskningsarbeten, måste du börja med ett eller flera forskningsfrågor om ett problem du vill lösa eller ett fenomen som du vill observera. Man måste också ta hänsyn till att byggandet av en modell kan vara mycket utmanande när inputs och outputs är många.

En enkel modell är inget mer än ett system av agenter och relationerna mellan dem, men det kan fortfarande uppvisa en djup nivå av komplexitet och därigenom ge insikt i verkliga fenomen genom observation i en kontrollerad miljö. Experiment i verkligheten kostar både tid och pengar och, som vi alla vet, så är verkliga miljöer och individer långt ifrån kontrollerbara. Använda ABM och Netlogo ger observatören eller forskaren utrymme att ändra parametrar inom modellen för att ta hänsyn till ett brett spektrum av omständigheter. Detta kan skapa värdefulla data för att utveckla ny hypotes om den genomförda forskningen.

Agentbaserad modellering har funnits i över 50 år, men sedan slutet av 1990-talet har det skett en kraftig ökning av användningen på många områden, vilket nämns både inom biologi och näringsliv och samhällsvetenskap. En av anledningarna är att data från exempelvis ekonomiska modeller, social nätverksanalys och geografiska informationssystem kan integreras i agentbaserad modellering. Det finns också sätt att göra ABM mer kraftfulla genom att använda maskininlärning. Men som tidigare nämnts är en enkel modell många gånger mer än tillräckligt för att observera ett fenomen. Människor observerar och förstår sin värld genom mönster och det är just vad Netlogo hjälper till med. Att bygga en komplex modell är långt ifrån lätt och förstå koncepten bakom kräver mycket forskning och arbete, men modeller är också byggda för att observeras och interagera med och även modifieras för att ge olika resultat - inte alla måste bygga modeller från början.

Sammanfattningsvis verkar det som Netlogo erbjuder en potentiell lösning för kartläggning av sociala nätverk. Det kräver en viss ansträngning att påbörjas, men borde motivera detta varhelst forskare har en tydlig hypotes som de vill modellera. Men i nuläget behöver vi förstå sociala nätverk bättre, och Netlogo erbjuder en metod som alla kan använda, även utan programmeringserfarenhet.

Netlogo kan ge ett värdefullt extra verktyg i alla forskningsverktygssatser som är utformade för att utforska sociala medier. Det förenklar också den lika viktiga processen att förstå de underliggande fördelarna med agentbaserad modellering. Varje forskare måste förstå vad agentbaserad modellering är och vad den kan uppnå innan de kan utforma en hypotes till modell. Netlogo erbjuder båda möjligheterna.