Antti Kangasrääsiö

Feasibility of Agent-Based Modeling and Simulation in Modeling Waste Value Chains

School of Electrical Engineering

Thesis submitted for examination for the degree of Master of Science in Technology.

Espoo 27.10.2012

Thesis supervisor:

> Prof. Kari O. Koskinen

Thesis instructors:

> Prof. Juha Kaila

> M.Sc. (Tech) Johanna Laaksonen

A! **Aalto University**
School of Electrical Engineering

Author: Antti Kangasrääsiö

Title: Feasibility of Agent-Based Modeling and Simulation in Modeling Waste Value Chains

Waste management policies have a large impact on how the waste management system behaves. However, deciding suitable policies may be difficult as the system may be very complex. Simulation models of waste management systems are important for the development of relevant waste management policies, as they allow the effects of these policies to be evaluated before they are actually implemented.

Waste value chain analysis is a new method for modeling waste management systems. The idea of waste value chain analysis is to model each decision maker acting in the waste management system as an individual actor that can make independent decisions. The effects of these decisions can be aggregated to model how the waste management system behaves as a whole.

This study evaluates the feasibility of implementing waste value chain analysis models using agent-based modeling and simulation. The motivation for using agent-based modeling and simulation is its ability to model distributed decision making using a variety of interacting agents.

This study constructs an experimental waste value chain analysis simulation model of the waste management of the Finnish daily consumables retail shops using agent-based modeling and simulation. It then analyzes the feasibility of the modeling paradigm for implementing these kinds of models based on literature sources, experiences gathered from the modeling project and opinions from subject matter experts.

The result of the study is that implementing waste value chain analysis models using agent-based modeling and simulation appears to be generally feasible. However, validating these kinds of models may be challenging, as a considerable amount of data needs to be collected.

Tekijä: Antti Kangasrääsiö

Työn nimi: Agenttipohjaisen mallinnuksen ja simuloinnin soveltuvuus jätehuollon arvoketjujen mallintamiseen

Päivämäärä: 27.10.2012   Kieli: Englanti   Sivumäärä:9+72

Automaatio- ja systeemitekniikan laitos

Professuuri: Automaation tietotekniikka   Koodi: AS-116

Valvoja: Prof. Kari O. Koskinen

Ohjaajat: Prof. Juha Kaila, DI Johanna Laaksonen

Jätehuollon säännöksillä on suuri merkitys jätehuoltojärjestelmän toiminnan kannalta. Tarkoituksenmukaisten säännösten laatiminen voi kuitenkin olla haastavaa, sillä jätehuoltojärjestelmä voi olla hyvin monimutkainen. Jätehuoltojärjestelmien simulaatiomallit ovat tärkeitä uusien säännöksien kehittämisen kannalta, sillä ne mahdollistavat säännöksien vaikutusten arvioinnin ennen säännöksien toimeenpanoa.

Jätehuollon arvoketju -menetelmä on uusi tapa mallintaa jätehuoltojärjestelmiä. Menetelmän ideana on mallintaa jokainen jätehuoltojärjestelmässä toimiva päätöksentekijä itsenäisenä toimijana, joka pystyy tekemään itsenäisiä päätöksiä. Näiden päätösten vaikutukset yhdistämällä on mahdollista mallintaa sitä, kuinka jätehuoltojärjestelmä käyttäytyy kokonaisuutena.

Tämän tutkimuksen tarkoituksena on arvioida agenttipohjaisen mallinnusmenetelmän soveltuvuutta jätehuollon arvoketjumallien toteuttamiseen. Agenttipohjainen mallinnusmenetelmä valittiin toteutustavaksi, koska se mahdollistaa hajautetun päätöksenteon mallintamisen käyttäen hyväksi erilaisia keskenään kommunikoivia agentteja.

Tässä tutkimuksessa toteutettiin kokeellinen jätehuollon arvoketjumalli Suomen päivittäistavarakauppojen jätehuollosta käyttäen agenttipohjaista mallinnusmenetelmää. Mallinnusmenetelmän soveltuvuutta tämänkaltaisten mallien toteuttamiseen analysoitiin kirjallisuuden, mallinnusprojektin aikana kerättyjen kokemusten sekä asiantuntijamielipiteiden perusteella.

Tutkimuksessa saatiin selville, että agenttipohjainen mallinnusmenetelmä soveltuu yleisesti ottaen jätehuollon arvoketjumallien toteuttamiseen. Tämän kaltaisten mallien validointi voi kuitenkin olla haastavaa, sillä se vaatii runsaasti dataa.

Avainsanat: agenttipohjainen mallinnus ja simulointi, jätehuollon arvoketju, jätehuolto

Preface

People have always had to deal with complex systems, but modeling them has never been easy. Agent-based modeling and simulation is an interesting new paradigm for modeling complexity that arises from a multitude of local decision makers with imperfect information and limited decision making power. On the other hand, society is producing more and more waste every year, so working towards having a good tool for modeling waste management systems is important. By combining the know-how from both modeling and waste management, this study has made a foundation for a new kind of waste management system model. I believe that one of the main reasons for the success of this project has been the open-minded attitude of the project participants concerning this kind of multidisciplinary approach. Working on the project has been an interesting journey to both agent-based modeling and simulation and waste management systems. I have learned a lot on agents, modeling projects, waste management, programming, technical writing and perseverance.

Making this thesis would not have been possible without my instructors Johanna Laaksonen and Juha Kaila, who worked with me on the modeling project as subject matter experts and also provided me with excellent feedback and ideas throughout the project. I would like to thank my supervisor Kari Koskinen for his excellent comments on writing the thesis and for arranging me the possibility to participate in the EcoBalance 2012 conference, where an article based on this thesis is presented. I would like to give special thanks to my father Juha for helping me with proofreading this thesis, and Ilkka Seilonen from Information and Computer Systems in Automation research group for his interest in my work and comments along the way. Also, I would like to thank Ekokem Oy for the scholarship that financed this thesis.

Additionally, I would like to thank all the people from Information and Computer Systems in Automation research group for these past years that we have shared together in the lab. I have been able to learn a lot during these years, and it has always been a pleasure to have good conversations in our coffee room Sumppimonttu. I would also like to thank all the people from the Guild of Automation and Systems Technology, who have been such a great company in all the numerous guild activities that I have had the pleasure to participate in during these past roughly five years. My thanks go also to my family and relatives, who have supported me during my studies. Finally, I would like to thank Elli for her love and companionship during these past nearly seven years.

Otaniemi, 12.10.2012

Antti O. Kangasrääsiö

Contents

Abbreviations

ABMS	Agent-based modeling and simulation
ACE	Agent-based computational economics
CAS	Complex adaptive system
CBA	Cost-benefit analysis
LCA	Life cycle assessment
MCDA	Multicriteria decision analysis
UML	Unified modeling language
VOMAS	Virtual overlay multi-agent system
WVCA	Waste value chain analysis

List of Figures

1 Introduction

1.1 Background and Motivation

Waste management is an important part of any product's life cycle, beginning from the product being discarded and ending in the materials being recycled back to circulation or deposition. Because there are many environmental effects concerning waste depositing, transporting, processing and other activities related to waste management, there is a need to regulate the industry so that the effects could be kept to the minimum while still ensuring the quality of waste management. Waste management is one of the most regulated sectors in modern society [3, ch. 1.4]. For example, the European Union Waste Framework Directive [4] defines some basic waste management principles that apply in the EU region and the Finnish Waste Act [5] defines how waste management should be organized in Finland. Modeling waste management systems is important for the development of various regulations, so that their effects can be analyzed before they are actually applied in practice. Waste management models are also important for following the environmental effects that result from waste management.

According to Morrissey and Browne [6], current models of waste management systems can be roughly divided into three categories: cost benefit analysis models (CBA) [7], life cycle assessment models (LCA) [8] and multicriteria decision analysis models (MCDA) [9]. However, there are many known limitations to these kinds of traditional models [6]. One of the limitations is the implicit assumption that there exists a single decision-maker who could direct the system. In reality, waste management systems are composed of a multitude of independent actors, who are primarily interested in maximizing their own benefit, instead of optimizing the system as a whole. Even though the policy makers have control over the regulations, the actors decide how they apply the regulations in practice. The existing models are useful, but a more versatile model that would take into account the distributed nature of decision making in the system could be used for more detailed analysis.

Waste value chain analysis (WVCA) is a new method for modeling waste management systems. In WVCA, the system is modeled by using a group of individual actors that produce, process, trade and deposit different types of waste and other commodities. The model can be used to simulate the system-level behaviors of the waste management system. The decision making of the actors is modeled using behavior models derived from various sources, such as production economics and existing regulations. Some of the actors have processes that they use to transform one type of material into others. In order to trade commodities, the actors negotiate trade agreements with each other. In this way, the economic, material processing and regulative aspects of the waste management system can be studied using a single model. The model can be used to simulate various system-level behaviors, such as material flows, transported ton-kilometers, processed tons and commodity prices.

The model has been so far only theoretical and is still under development by the Aalto University Department of Civil and Environmental Engineering. Further study and development of the model requires a method for implementing it. In

this study, an experimental model is implemented using the agent-based modeling and simulation (ABMS) paradigm [10]. It was chosen because the theoretical model and the modeling paradigm are both based on groups of heterogeneous independent decision makers. It was expected that this kind of complex system could be modeled using ABMS, as promising results had been reported from reasonably similar modeling projects, such as modeling deregulated energy markets [11].

1.2   Scope and Contributions

The focus of this thesis is to assess the feasibility of implementing WVCA models by using the ABMS paradigm. The feasibility is examined from the points of view of model design, implementation, verification, validation, credibility, experimentation, analysis of results and comparison to alternative methods.

Experiences of using the ABMS paradigm were gathered by implementing an experimental simulation model. The waste management of Finnish daily consumables retail shops was chosen as the reference system, as it is a sufficiently complex, yet manageable system to model. Additionally, the experimental model was presented to a group of experts from fields of waste management and modeling, and written anonymous feedback was collected.

The modeling paradigm was found to be adequate for this modeling task. First, the structure of ABMS models corresponds well with the WVCA model. Second, the modeling method is very versatile and allows the implementation of various different features and behavior models. Third, the model can be used to generate a large amount of data, allowing versatile analysis and comparison of different scenarios.

Also some challenges were identified. First, the model requires a large amount of data for parameterization and validation, which may not always be available. Second, constructing credible behavior models for the model actors can be challenging if the actual behavior is complicated. Third, the simulation runs are computationally intensive, which may make studies impractical unless sufficient computing power is available.

1.3   Structure of the Thesis

The theoretical foundations of the different methods, models and systems are described in sections 2 to 5. Section 2 focuses on computer simulation modeling, section 3 on the ABMS methodology along with some related fields of study, section 4 on the WVCA model and section 5 on the system to be modeled.

In section 6, the practical part of the project is described, with details on the project workflow, model definition, tools used and implementation. In section 7, the feasibility of the model is analyzed. In section 8, a summary of the thesis is given along with suggestions concerning the further development of the model.

2 Computer Simulation Modeling

2.1 Overview

Computer simulation means imitating real or imaginary systems using a computational model of the system under study. Computer simulation is a practical way to approximate how a system behaves without the need to actually implement the system. Computer simulation can be used, for example, to analyze a system's design before the actual implementation, to study the behavior of a system without needing to conduct possibly expensive experiments, or for educational or demonstrational purposes.

To perform computer simulation, a computational model of the system needs to be constructed. This model can usually be derived from known or presumed system properties and behaviors, which are then transformed into mathematical functions and algorithms that are implemented using a programming language. Constructing these kinds of models is called simulation modeling.

There exists multiple simulation modeling paradigms, such as Monte Carlo models [12], discrete event models [13] and hybrid models [1, ch. 1]. Each of these has its own advantages, so the modeling paradigm should be chosen to best suit the situation. Determining the intended use of the model, needed level of detail and other key aspects can help in choosing the best one [14].

Simulation modeling is already a mature technique. Various handbooks, such as *Simulation Modeling and Analysis* by Law and Kelton [1], exist for in-depth reference on the subject. In this section some important concepts related to this study are reviewed. A general simulation and modeling project workflow is introduced and the verification, validation and credibility of models are discussed together with some general considerations that are relevant in simulation modeling.

2.2 Modeling Project

A simulation modeling project can be divided into different phases, such as planning, implementation and analysis of the results. The workflow of the project is usually iterative, meaning that the earlier phases of the project may be revisited when needed. In this subsection, a general modeling project workflow, adapted from [1, ch. 1], is described.

The workflow of a simulation modeling project is illustrated in figure 1. The project starts with a planning phase, where the objectives of the project and the specific issues to be solved are decided. These specifications should define, for example, which system and which behaviors are to be modeled, what is the intended use of the model and what kind of data is to be used for parameterizing and validating the model.

The next phase is the definition phase, where the requirements for the simulation model are formulated and relevant data is collected. These requirements can be less detailed at first, and refined after a simple model with a few basic features has been implemented. The collected data includes both parameterization and validation
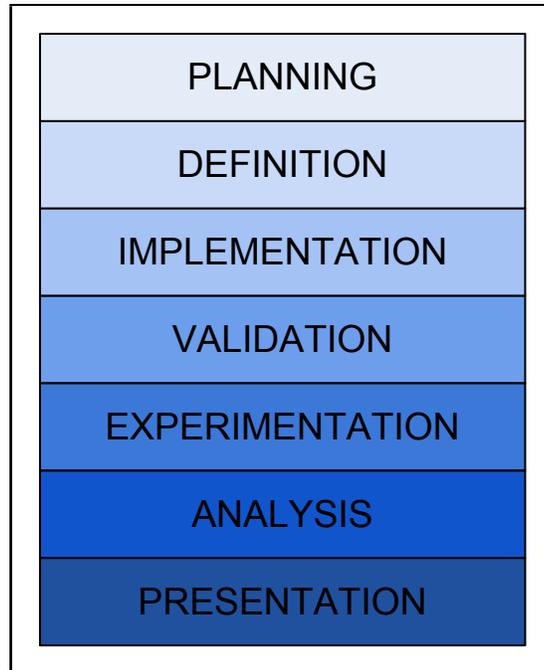
Figure 1: A general modeling project workflow, adapted from [1, ch. 1].

data. In order to clarify the requirements, the modeler should discuss with the model users and subject matter experts. After the requirements are finished and the data collected, it is a good practice to reflect on the model specifications, and determine if they still agree with the actual system. If there are some problems with the model, it can be redesigned before the implementation begins.

If the model specifications appear to be sufficient, the next phase is the implementation phase, where the model is programmed. Model verification should go hand in hand with the implementation, meaning that the software should be tested during implementation to make sure it works as specified.

After the model has been implemented and verified, the next phase is the validation phase, where the model is validated. Validation can be done objectively, usually by comparing the simulation results with the validation data, or subjectively, usually by the help of subject matter experts. If the simulation results are acceptable and the model appears to be valid, the project can move to the next phase. However, if the simulation results are not acceptable, the project should return to the definition phase to refine the model.

Once the model is validated, the next phase is the experimentation phase, where experiments are designed and executed using the simulation model. When designing the experiments, various aspects should be taken into account, such as the limits of the model validity, sensitivity of the model to variations in initial conditions, reliability of the data and the type of analysis used. Once the experiments have been designed, they can be executed to get the simulation results. After that, the next phase is the analysis phase, where the results from various experiments are

analyzed and conclusions are made.

After arriving to conclusions based on the simulation study, the project moves on to the presentation phase, where the results of the study are presented, for example, to relevant parties, or the results are used in another project.

2.3   Verification, Validation and Credibility

The most important requirements for a modeling project are to create a model that works as specified, produces behavior that is similar enough to that of the real system and is accepted by the users as correct and usable. Equivalently it can be stated that the model should be verified, validated and credible. In literature, there exists slightly varying definitions to these terms. In general, verification refers to ensuring that the simulation model has been implemented correctly, whereas validation refers to ensuring that the model functions similarly to the real system. Credibility, on the other hand, refers to the users' acceptance of the model, which depends on, for example, the understandability of the model and the suitability of the model for the users' needs.

There exists various techniques for verification, validation and ensuring credibility. However, there is no universal method to choose the best one to use, so the decision is usually done case-by-case. In this subsection, a few methods that are the most relevant to the project at hand are presented. For more detailed analysis on the subject, see for example [1, ch. 5] and [15].

Verification

A model is said to be verified when the model implementation works according to specifications. There are many complementing ways to verify a model, and usually more than just one way is used for verification. The availability of the methods can depend on the programming language and environment used. Using a specialized simulation language, such as Simulink, usually reduces the need for verification, as the restricted structure of the language leads into simpler code with fewer possibilities to have errors. Conversely, when using a general-purpose programing language, such as C++, the need for verification is usually greater.

For verification, one option is to use test programs that execute the model, or parts of it, using various input patterns and check that the model output is as specified. Verifying modules of the program independently is known as unit testing. However, this usually leads to black-box testing, meaning that only the apparent functionality is tested, not the internal working of the module. Unit testing is usually followed by integration testing, which means testing that the unit-tested modules work together as specified.

Another way to verify the model is to interactively debug the program when it is being executed. In this way all the program variables can be observed on every step of the program execution, which allows the verification of the internal workings as well. However, this approach is usually very time consuming if there are a lot of variables or different scenarios to consider, therefore it should be generally restricted

to only the most important parts of the program.

A third way to verify a model is to graphically visualize the model states. This is an intuitive way to observe the model execution, even though there might be some aspects of the model that cannot be easily visualized, such as high-dimensional data or complex algorithms. It should be noticed that the visualization environment in itself should also be verified.

Validation

A model is said to be valid when experiments made with the model yield results reasonably similar to those that the actual system would yield if experimented likewise. The scope in which the model has to be valid and the needed accuracy are defined by the purpose of the model. Therefore, the methods used for validation should be chosen to best match those purposes.

Validity can be divided into three levels: replicative, predictive and structural validity. Replicative validity requires that the model is able to replicate observed behavior, meaning that the model is able to behave similarly as the actual system is known to behave. Predictive validity requires in addition that the model is able to predict the behavior of the actual system in situations that have not yet been observed. Structural validity is the strongest level of validity, and it requires, in addition to replicative and predictive validity, that the structure of the model works in the same way as the structure of the actual system. [16, ch. 2]

There are both objective and subjective approaches to validating a model. Objective approaches rely generally on statistical analysis of the model behavior, whereas subjective approaches rely on expert knowledge on how the system should behave. There are cases where some methods of validation may not be available. For example, if the system does not actually exist, comparing the model behavior against actual behavior is not possible. As with verification, many complementing ways are generally used for validation as well.

If there is sufficient data available of the system, one way to objectively validate a model is to compare this data to the model's behavior. By using statistical methods, it can be quantified how well the model imitates the known behavior of the system. The sensitivity of the model to parameter variations can also be used in validation. Generally, the sensitivity of the model should be similar to that of the real system. With sensitivity analysis, it can be additionally determined which parameters have to be set up accurately for reliable behavior of the model (i.e. those parameters that have the largest effects on behavior).

One approach to subjective validation is to ask experts whether the model of the system and its behavior seem to be reasonable. Another approach is to ask the experts whether they can distinguish between the system and model outputs, without being told which one is which. Subjective approaches generally rely on the experts' understanding about the actual system, and therefore, they should be used with care when the actual system behavior is not well understood even by experts.

Validation may also include model calibration, which means setting unknown or uncertain model parameters to such values that the model behaves as intended.

Calibration can be done manually or algorithmically. However, when calibrating a model, it should be taken care that the model is not over fitted, meaning that the model is calibrated to replicate only a particular set of examples instead of the general behavior. The possibility of over fitting can be reduced by using separate sets of data for calibration and validation. [1, ch. 5]

Credibility

The credibility of the model depends not only on the model, but also on the user and the interaction between the user and the model. In general, credibility refers to how believable, reliable, understandable and usable the model is from the viewpoint of its user. Lack of credibility can result, for instance, from difficulty of understanding the model's behavior, unsuitability of the model for the desired use or insufficient validation and verification.

To ensure model credibility, it is important for the modeler and the model users to have similar views concerning the model. This can be ensured by regular interaction throughout the modeling project. Points to be agreed on include, for example, the model requirements, intended use and required level of validity. Documenting these in an understandable manner is one way to make the project requirements clear to all parties.

Making the model and the modeling project transparent to the user can also increase the credibility of the model. This can be done, for example, by documenting the model structure and the verification and validation steps, and by having the users participate in the planning and validation of the model.

2.4   General Considerations

Simulation modeling can be a complicated project. For example, the modeler might not be an expert on the subject to be modeled, the model users might not be experts on modeling, or the system under analysis might be complex with many possible ways to construct the model. Even though modeling projects can be very different from each other, there are some general considerations that apply for most of the projects. In this subsection some of these general viewpoints are discussed in more detail. Further discussion on general considerations in modeling can be found, for instance, in [1, ch. 5] and [17].

When constructing a model, a choice has to be made concerning what aspects of the actual system are to be included in the model and what are to be left out. The model should be designed with the specified uses in mind so that the appropriate level of detail can be decided. The modeler should discuss with the model users to clarify the requirements and consult subject matter experts for insight into the key areas of the actual system. Generally, the model does not need to have one-to-one correspondence with the actual system in order to yield acceptable results. As the complexity of the model increases, so does also the effort needed to implement, verify and validate the model.

The purpose of the model is to be a tool for analysis and decision making.

Thus, an important part of model construction is its usability and suitability for the designed use and users. It needs also to be decided what kind of control the user has over the simulation and what kind of feedback does the user get from the model. For instance, the control could be in form of user adjustable parameters or datasets, and the feedback could be visualizations or statistical analyses. To make sure that the model matches the needs of its users, the modeler should also involve the users in designing the model requirements and user interfaces.

The model construction should be gradual, starting from a simple conceptual model and adding up features until the model finally meets the design requirements. This enables the modeler to quickly create an initial model in order to study the overall feasibility of the model structure. If the initial model appears to have major issues, it can be discarded and redesigned without any major losses from implementing the whole model. Also, this method allows the modeler to obtain intermediate results throughout the development process, which can be valuable for other projects.

3 Agent-Based Modeling and Simulation

3.1 Overview

ABMS is a paradigm for modeling systems with distributed decision making. The model is composed of a multitude of independent agents that are situated in an environment. The dynamics of the model result from the agents interacting with each other and the environment, based on different behavior models. The model is derived by choosing representative agents from the system and modeling their behavior, as well as the dynamics of the environment. The fundamental assumption of ABMS modeling is that the different entities and their behaviors can be credibly modeled [18], and that by allowing these entities to iteratively interact with each other the resulting behaviors are similar to those of the actual system. ABMS is an intuitive way to model systems where decision making is distributed and the behavior of entities that make the decisions is relatively well understood.

ABMS originates from the study of complex adaptive systems (CAS) [19]. Some of the first widely known agent-based models were the Boids model by Reynolds [20] and the Sugarscape model by Epstein and Axell [21]. The Boids model was used to simulate the flocking behavior of groups of animals that moved around based on simple rules that took into account the movement of the agent's neighbors. The Sugarscape model was used to model groups of social agents that moved around a grid with different amounts of sugar in every cell. The agents could interact with each other and the environment in different ways, such as by metabolizing sugar, reproducing, and transferring information with each other. Nowadays ABMS has been applied to model various complex systems, such as deregulated energy markets [11], economy in Europe [22], granuloma formation during tuberculosis [23] and effects of biological warfare [24].

In essence, ABMS is a microsimulation paradigm, where the system-level behaviors are modeled using lower level objects. This can be contrasted with macrosimulation, where the system is modeled using system-level objects. As an example, traffic flow in a freeway can be modeled in different ways. If the traffic would be modeled as a continuous flow, with different vehicle densities and velocities in different points of the freeway, it would be a macrosimulation model, as system-level properties are being used to describe the system. However, if the vehicles would be modeled individually, so that each of them would have certain properties like speed, position and route to follow, it would be a microsimulation model, as an individual vehicle is essentially a low level object compared to the traffic flow. The benefit of microsimulation is that it makes it possible to study the relations between the lower and higher level behaviors in the system. However, this usually increases complexity as the level of detail is greater.

In ABMS the term 'agent' is defined in slightly different ways in different sources [18]. In general, an agent is defined quite loosely to be a model component that can make independent decisions and change its behavior in response to its past experiences [18]. It should be noted that the term 'agent' has also uses outside ABMS. For example, software agents are autonomous software components used in

software development [25] and robotic agents are robots that are able to plan their actions and also learn from them [26].

This section starts with a review of two related lines of study: complex adaptive systems (CAS) and agent-based computational economics (ACE). CAS is included to illustrate the way it is natural to think about systems when modeling with ABMS, and ACE is a line of study focused on applying ABMS to economic problems, and it is included as it is related to the actual model we are constructing. After these reviews, a general theoretical description of ABMS models is given, followed with notes on design, implementation and general considerations when using ABMS.

3.2   Complex Adaptive Systems

ABMS originated from the study of complex adaptive systems (CAS) as a set of ideas, techniques and tools that enabled creating simulation models for the purposes of analyzing CAS [10]. Although not directly related to this project, the general concepts from this line of study provide useful insights into the modeling viewpoint of ABMS.

A system can be said to be a CAS if it exhibits coherent behavior under change [27], roughly meaning that the system can adapt by itself to survive in variable conditions. For example, the electricity market can be said to be a CAS, whereas a traditional physical electric grid cannot. The market can adapt to various changes in supply and demand, find suitable market prices for electricity and constantly evolve towards better market strategies by rewarding those companies that can outperform their competitors. On the other hand, the physical electric grid has only limited ability to adapt to different changes in its environment without outside regulation, and it has no ability to learn from its past performance by itself.

Constructing tractable models of CAS was difficult by using traditional modeling methods, such as linear algebra or statistical methods, which lead into the development of ABMS. As CAS are composed of large amount of heterogeneous adaptive entities that have different ways to interact with each other and the environment, these entities were named as agents and used as the building blocks of the model.

Features of a CAS and their use in modeling

Holland has defined 'seven basics' that are common to all CAS: aggregation, tagging, nonlinearity, flows, diversity, internal models and building blocks [27]. They represent the basic common properties and mechanisms that can be found in all CAS. They are explained next, with comments on their relevance to modeling.

Aggregation means that similar properties and entities can be aggregated into classes, and members of a particular class can be treated as equivalent. This helps modeling such systems, as we can derive useful abstractions thanks to these classes. On the other hand, aggregation also means that even though the actions of a single agent are simple, by aggregating the simple behaviors of all the agents in the system it is possible to create complex system-wide behavior.

Tagging refers to the existence of certain types of specific patterns that enable

agents to interact selectively with their environment. For instance, animals can recognize other animals of the same species based on different cues, such as coloring, sound and smell. Finding the relevant tags helps the modeler in choosing relevant parameters and types of interaction for the agents.

Nonlinearity refers to the fact that the processes in CAS are usually nonlinear. Many of the variables that characterize the system are usually dependent on each other, which generally makes analysis difficult. However, by making a constructive model of the system, the analytical solution does not necessarily have to be solved in order to replicate the results.

Flows refer to the movement of information and resources in the system. In a commercial market these could be things like raw materials, products and business offers. Identifying the key flows makes it easier to specify and parameterize the interactions that agents have with each other and the environment.

Diversity refers to the heterogeneity of the agents and their strategies. Even though agents might be originally similar, by adapting to different situations, diversity is formed. By examining the diversity inside an agent class, various relevant degrees of freedom for the agents can be inferred.

Internal models refer to the ability of agents to make predictions of their future state based on the current state. This internal model should also be able to change in response to the agent's experiences. This means that the agents should be able to learn, either directly from experiences (similarly to humans) or indirectly by natural selection (similarly to evolutionary algorithms). By understanding the internal models of agents, the decision processes can be refined to better represent those of real agents.

Building blocks refer to the ability of agents to construct internal models of a complex world using simpler building blocks. On the other hand, building blocks also refer to the layer-like structure of CAS, where an aggregation of lower level agents can be seen as a higher level meta-agent. For instance, a military unit agent could comprise of rifle team agents, and a military company could comprise of various unit level agents. Again, by identifying the relevant building blocks, useful abstractions can be derived for the model.

Features of a CAS agent

Holland identifies three major components that a CAS agent needs: a performance system, a credit assignment algorithm and a rule-discovery algorithm. They are explained next, also with comments on their relevance to modeling.

Performance system is the part of the agent that generates its current behavior. It comprises of a set of detectors that collect information about the environment, a set of effectors that can interact with the environment and a set of rules that decide what to do and when. This implies that the agent should have clear definitions concerning what information it can receive and from where, and what actions it can take and when. The agent should also have a clearly defined model for making decisions.

Credit assignment algorithm is responsible for evaluating the current perfor-

mance of the agent and indicating the behaviors that lead into better results. This implies, that the modeled agent should be able to evaluate its own actions, and decide which behaviors lead to good and which to bad results.

Rule-discovery algorithm is responsible for refining the behavior caused by the performance system, considering the evaluation done by the credit assignment algorithm. This implies that the agent should be able to make meaningful changes to its behavior based on previous experiences, to adapt. This also implies that the performance system should be changeable or parameterized in order to permit this adaptation.

3.3 Agent-based Computational Economics

As the system being modeled is fundamentally economy-driven, agent-based computational economics (ACE) offers some important concepts that can be used in the modeling project. The ACE line of study has four main goals: to understand why certain market behaviors arise and persist in certain situations, to create better economic strategies for the future, to get a deeper understanding of the economic theory and to develop better tools for economic study [28, ch. 16].

Classical micro-economic theory comes with many assumptions, such as that economic agents are rational and constantly optimizing their behavior, and that they are all identical in their properties [18]. There are also classical pricing mechanisms, such as the Walrasian auctioneer [29], which can be used to determine the theoretic prices of goods in a market. However, these kinds of classical models and assumptions are simplified and do not directly represent how real-world economies work through various procurement processes [28, ch. 16].

The driving force of an ACE model is the procurement process [28, ch. 16]. A procurement process consists of economic agents identifying what goods and services they wish to buy and sell, and at what prices and quantities. The agents must identify potential trade partners and send offers to buy and sell accordingly. Received offers must be compared and evaluated, suitable offers can be accepted and long term commitments must be managed. These processes vary in complexity and details, depending, for example, on the type and amount of commodity traded, the competition on the market and the common history with the trading partner.

In classical theories the procurement process is simplified to enable the construction of analytically tractable models. However, this also leads the theoretic results to differ from the reality [28, ch. 16]. With agent-based methods, the individuals behind the procurement process can be modeled directly with agents, which allows the modeler to make less generalizations and assumptions, leading into a more accurate model.

ACE models can basically be seen as a subset of ABMS models focused on economy. Agents have different utility functions, which they try to maximize and different procurement strategies which they use to make deals with other agents. The modeled economic system normally develops solely based on agent interactions, without external interventions from the modeler. The agents have limited and variable amount of information and rationality in use for making the decisions, which

is alike in the real market.

Economic theory is usually focused on finding equilibria in certain types of situations. In ACE models, equilibria may not always exist, as they are only possible emergent behaviors that may arise from the agent-level interactions. The focus of ACE is to study when and where the possible equilibria are formed and to study the initial conditions which lead the system to converge. One example of an equilibrium is a state, where stable trade networks have formed. Another example would be a state, where the market prices for certain commodities have found a stable level. Various other concepts for equilibrium also exist, and they are not necessarily dependent on each other, meaning that the system can be in equilibrium in one sense, but not in the other.

3.4 Structure of ABMS Models

An ABMS model consists of various structures, such as entities, contexts and schedules. The entities can have different properties, behaviors and interfaces to other entities. A context is a container for groups of entities, which act in the order given by the schedule. In this subsection, the parts of an ABMS model and their general properties are introduced. More details on the subject can be found for example in [10].

An entity in the simulation can be roughly defined by an interface, a set of attributes and a model of behavior. Generally, these are identical in structure for a certain type of entity. The interface represents the ability of the entity to interact with other entities in the model. The set of attributes represents the state of the entity, which can change as the entity experiences different events. The model of behavior represents the logic that determines which actions the entity will take and when. It is usually implemented as a set of functions and algorithms, parameterized by the attributes of the entity. The entities can generally be divided into agents, environmental entities and simulation function entities.

Agents are entities that have goals to achieve and methods to adapt to the current situation. The goals of the agent can be defined in many ways, but generally they represent the preference of some situations over others. Goals can be defined either explicitly, for example, by defining a mapping from the agent's situation to a level of utility, or implicitly, for example, by defining the preferred course of action in a certain situation. One way to define adaptation is, that the agent can use its past experiences and information currently available to it to make meaningful decisions in order to pursue its goal states in different situations.

Environmental entities are generally preprogrammed in their actions. They do not have either goals to achieve or ways to adapt to the current situation. They represent the structure of the agents' environment, and can restrict or enable the behavior of the agents. For example, a tree entity could enable an animal agent to feed or climb to safety, whereas a boulder entity might restrict the directions of movement of the animal agent.

Simulation function entities are not usually a part of the actual system model, but have other purposes, such as gathering information from the other entities,

processing the gathered simulation data, reading and printing data files, validating and verifying entities run-time, acting as interfaces between other simulation models and visualizing the model. These entities basically provide the user the practical means to interact with the model.

A context contains a group of entities which act according to a common schedule. The group of entities that an entity can interact with is called the neighborhood of the entity. The neighborhood usually consists of other entities near the entity's location. The distance between entities can usually be calculated from the used topography or read from a distance matrix. The model can have more than one context, or nested contexts, provided that the interactions between different contexts are defined.

As the behavior of the entities is simulated on a computer, where usually only a limited amount of parallelism is possible, there has to be a scheduler that determines the order in which the entities act. The scheduler contains a logic for determining the entity next in turn and an interface to signal the entity that it is allowed to execute. The interface may have different signals for different types of execution. For example, in a football game simulation the scheduler may allow each player in turn to plan its move and then to move, or allow each player first only to decide how they will move and then go through them again in order to allow them to move. The first example represents an unsynchronized and the second one a synchronized simulation of the game.

3.5 ABMS Modeling Project

In section 2, the general development project of a simulation model was introduced. The workflow of an ABMS modeling project does not differ from the one presented, although using ABMS models does raise some considerations during the design and implementation of the model. In this subsection these considerations are discussed in more detail.

When modeling a complex system, the system can be difficult to conceptualize and to translate into a model. To help this, one method is to answer a set of relevant questions regarding the major design decisions [10]. The following set of questions provides an example of this kind of set. They have been adapted partly from [10].

- What is the specific problem and the specific questions that the model should solve?

- What are the major benefits of using ABMS instead of other modeling paradigms for this system?

- Does the system have multiple logical or functional layers? What kind of layers there are and which ones are relevant?

- What kinds of time constants are relevant in the system?

- What are the relevant entities in the system?

- What entities can be aggregated together and treated as equal?

- What are the active entities or decision makers in the system?

- What are the entities that have goals or adapt?

- What entities should be represented as agents in the model?

- What are the decisions the agents make?

- What kind of internal models of the world do the agents have?

- What agent behaviors are relevant to model?

- What actions do the agents take and when?

- What kind of static and dynamic properties do the agents have?

- What is the environment the agents exist in?

- What kind of entities there are in the environment?

- How are the agents' neighborhoods defined?

- How do the agents interact with each other?

- How do the agents interact with the environment?

- How do the environmental entities interact with each other?

- What kinds of flows there are in the system?

- What kind of data there is about the system?

- Where does the data come from and how reliable is it?

- How can the model be validated?

- How credible could this model be?

By answering to a question set, such as the one presented, the modeler should get an idea of the major design decisions, such as what kinds of agents there are in the system, how they behave and in what kind of environment they exist.

As the system being modeled is usually complex, it would seem that the simulation model would also have to be equally complex. However, the entities in the model should be kept simple, and let the complexity arise from their interactions. By this way the model and the implementation can be kept relatively easy to understand, while still being able to replicate complex behavior on the system level. More analysis on simulation model complexity can be found for instance in [30].

When implementing the model, model development should generally be gradual. One way is to proceed in an iterative manner, starting from a simple model, and

then gradually implementing more features until the model has reached the intended requirements [10].

The model can be implemented using basically any general programming language and development environment. However, an ABMS toolkit can be useful, as they include program libraries and tools for modeling. By this way the main effort can be focused on implementing the actual model instead of the supporting functionality. Toolkits vary in the amount and type of libraries, features and supported programming languages. Repast Simphony [31], MASON [32] and NetLogo [33] are popular ABMS toolkits used nowadays.

3.6 Considerations in ABMS Modeling

Every type of modeling technique has its strengths and drawbacks. In this subsection, some general considerations that should be taken into account in ABMS modeling are discussed.

ABMS is a constructive paradigm, meaning that the model is constructed by defining how the individual parts of the system behave, but without explicitly defining how the system should behave at a higher level. The behavior of the system is a result of the interactions and behaviors at the lower level, meaning that the high-level model is implicitly defined by the definitions of the low-level entities.

The benefit of this kind of model structure is, that complex systems can be modeled without having to define explicitly how the system should behave on the system-level. This allows the system to be modeled, even though an analytically tractable model would be difficult to be constructed.

As there are usually a large number of agents and interactions between them, a large amount of data can be collected from the simulation. The large amount of data allows versatile analysis of the simulation results. Many different system-level behaviors might be observable from the data, and these behaviors might not be limited to those the model was primarily intended to replicate.

On the other hand, the constructive model structure may create complex behaviors, which may be challenging to analyze. For example, the model may be chaotic, meaning that it does not converge into any stable state, or it may be very sensitive to parameter and initial values, meaning that the results may not be continuous with respect to model parameters. [34]

Validating complex behavior requires the use of different techniques. For example, statistical methods, time series analysis and sensitivity analysis may be required. The validation may also require a large amount of different types of data to be collected, which may be problematic. For example, accurate data on the agent level may not be available or it may be expensive to collect. [34]

There are usually a large amount of parameters in the model, and therefore, there is a risk of over-parameterization. This means that the model may have more degrees of freedom than are needed. Because of this, there is a greater risk of overfitting the model, meaning that the calibrated model may be just tuned to represent the particular calibration dataset rather than the general behavior of the system. If the number of degrees of freedom is large, it may be possible that the model can be

calibrated to replicate almost any kind of behavior. Because of this property, the model may be impossible to be falsified, as it may be difficult to say for certain whether the model is correct or not. [34]

One method for verifying and validating ABMS models is VOMAS (Virtual Overlay Multi-Agent System) [35]. In the method, VOMAS agents are included the ABMS model. VOMAS agents can monitor the different simulation entities and general trends, and report if they notice that some constraints have been broken. This allows the verification and validation of the model during runtime. However, this requires that the model requirements and desired behavior should be specific enough to be programmable as constraints into the VOMAS agents. It should be noticed that also the VOMAS agents should be verified in some manner.

4 Waste Value Chain Model

4.1 Overview

Waste value chain analysis (WVCA) is a new approach for modeling waste management systems. The motivation for this model was already discussed in section 1, therefore this section will focus on describing the model structure. The model presented here is not yet final, as the model is still under development. For this reason, no references exist at this time. The model structure is described as it was during the modeling project.

In general, the model consists of actors that produce, process and deposit different types of commodities. Actors represent the various parties in the waste management system, such as communities that generate waste, companies that collect and process waste and landfills that deposit waste. Commodities, on the other hand, represent the different types of waste and recycled products that are traded between actors. The actors have different models of behavior that direct which commodities are traded with whom and at what price. These properties are elaborated in the following subsections. A simplified visualization of the model is shown in figure 2.
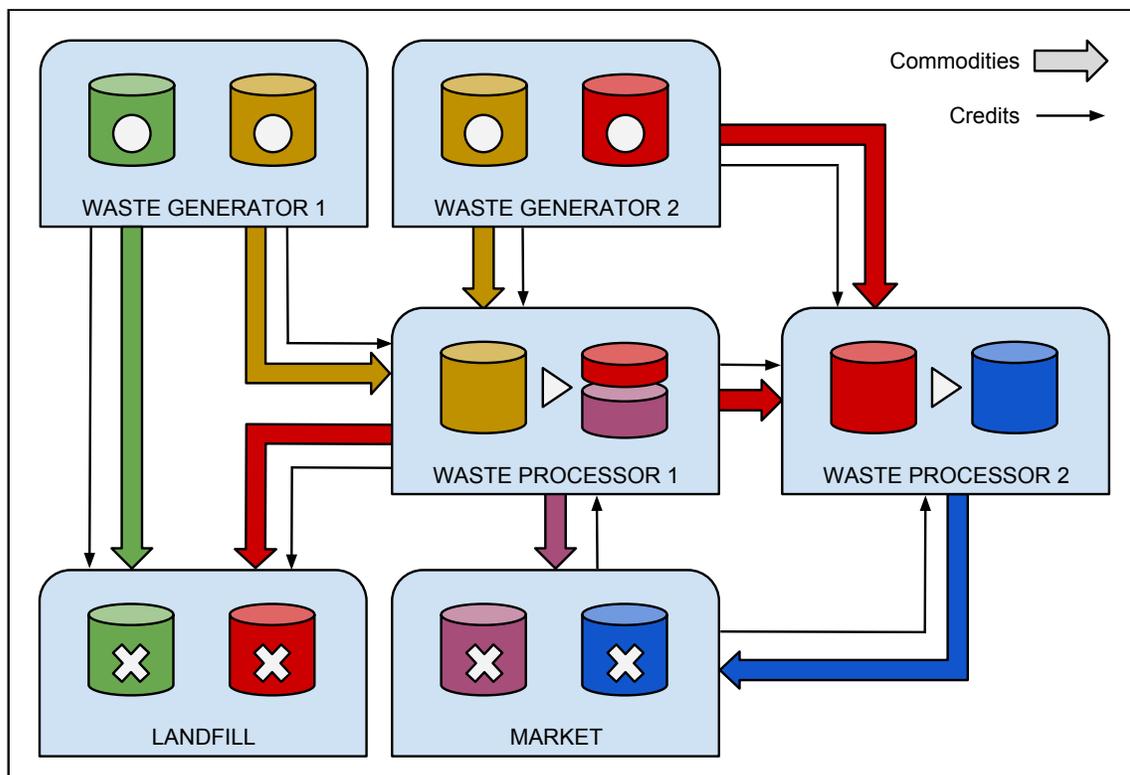

Figure 2: A simplified example of a waste value chain.

4.2 Commodities

Commodity can be defined as something that is traded and has value (which can also be negative). In the field of waste management, commodities include different types of waste, such as biowaste, mixed waste and plastics, but also products that are derived from waste, such as bioproducts, electricity and recycled materials.

In the model, commodities are handled in batches. A batch has two parameters: type and amount. The type of the commodity determines its properties, such as how it can be processed and which actors can receive it. The amount of the commodity determines how large amount of the commodity there is in the batch. The unit of the amount is determined by the type of the commodity. For example, mixed waste could be measured in tons whereas electricity could be measured in MWh. A batch of commodity can be used to represent, for example, the commodities stored by an actor or a shipment sent from one actor to another. Batches of the same type can be arbitrarily divided and combined with each other.

In real life, waste classified under a certain type can have a very heterogeneous composition. For example, mixed waste can include anything from glass and biowaste to metals and paper. Waste composition can also vary by batch. However, some aggregation has to be made to make batches comparable with each other. In the model, all batches of the same type are assumed to have homogenous composition. Thus, a ton of packed biowaste produced by one actor in the model is assumed to be identical to a ton produced by another actor.

When traded, a batch becomes a part of a shipment, which has additional properties, such as supplier, receiver, distance transferred and value. Supplier and receiver are determined by the parties in the transaction, distance by the locations of the parties, and value by the agreement the parties have with each other. When an actor sends a shipment, a batch is taken from its storage and attached to the shipment. When the shipment is received by the other actor, the batch is added to the existing storage.

4.3 Actors and Processes

Actors in the WVCA model represent the decision makers in the waste management industry. The level of abstraction is limited to company-level decisions, such as trade agreements and process usage. An actor is situated at a site, where all the company functions are assumed to be. Actors are divided into different types, such as landfills and waste treatment companies. Actors of the same type have similar behavior model, but different properties, such as size and location.

An actor is composed of a behavior model which is responsible for making all the decisions, a storage which contains all the commodities the actor owns, a mail room which contains all the pending messages the agent has received, an archive which contains all the relevant data the actor has stored, and other properties, such as assets, neighborhood and reputation. Some of the actors have also a process which allows the actor to process commodities. The properties of the actor define, for example, how it will act, what commodities it will trade and produce and who

it will interact with. A simplified model of an actor is shown in figure 3.
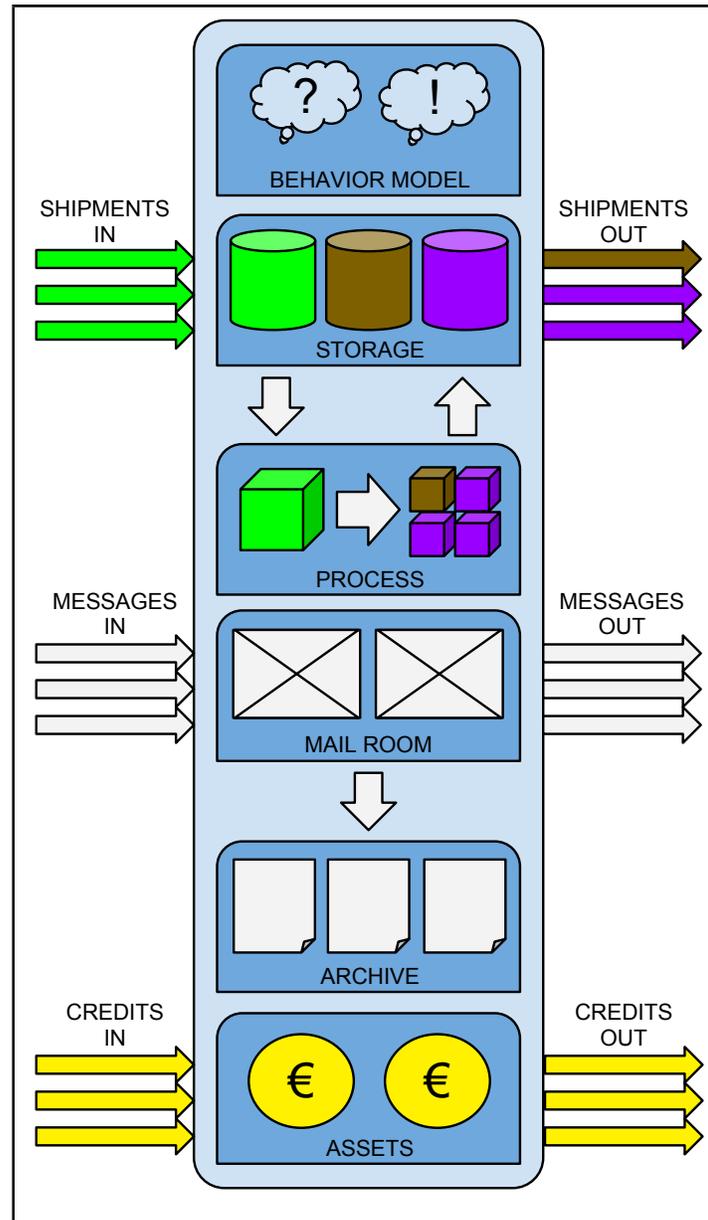
Figure 3: A simplified model of an actor with a process.

A process is an important part of many companies in the waste management industry. A process is something that can be used to transform commodities into other commodities. For example, waste incineration plants and composting plants are processes common in the waste management industry. Waste incineration plants could be used, for example, to transform mixed waste into electricity, district heat and ash. In the model, processes have a list of commodities they can process, a model for determining the process end products and expenses, and capacity limitations of the process.

4.4 Messages

The actors are able to communicate with each other using messages. Messages are used for making business deals, such as agreeing on the price of trading some commodity. The format of the messages is the same for all actors, so that all the actors are able to communicate with each other if needed.

In the model there are three types of messages: requests, offers and agreements. A request message indicates that an actor would like to receive an offer concerning the trade of some kind of commodity. An example of a request would be: 'I would like for you to accept mixed waste from me'. After receiving a request, the message should be answered with an offer message that indicates the terms concerning that trade. For example: 'I will accept mixed waste from you if you pay me 50 euros per ton'. An offer can then be either accepted or rejected. An offer is accepted by sending an agreement message, for example: 'It is agreed that you will accept mixed waste from me and I will pay you 50 euros per ton for it'. The messages have other parameters as well, such as the amount that is to be traded and the duration of the agreement.

4.5 Behavior Models

The behavior of the actors is decided by the behavior model. These models vary from simple to complex, depending on the actor. The models can be derived from different sources, such as economics, behavioral sciences and subjective observation. The only limitations for the model are, that it has to be specific enough to be implementable and to obey the given regulations.

As the system being modeled is fundamentally an economic system, the behavior models are focused on making economically rational decisions, such as preferring actions that are more profitable. However, the actors may also have goals that do not directly translate into monetary gain, such as preferring recycling of waste over deposition.

4.6 Time in the Model

In the model, time is divided into periods of equal length, which are evaluated in order. The evaluation of a period is divided into phases, each with a certain purpose. After a phase has been evaluated, the next phase is chosen based on the results of the previous phase. After the last phase is successfully completed, the first phase of the next period begins. There are a total of eight phases in one period. The phases are illustrated in figure 4.

The first phase is the generation phase, during which appropriate amounts of waste are generated to waste producers. After the generation phase comes the request phase, during which the actors are able to send request messages to all the other actors in their neighborhood. After the request phase comes the offer phase, during which the actors process the request messages they have received and respond to them with appropriate offer messages. After this comes the agreement
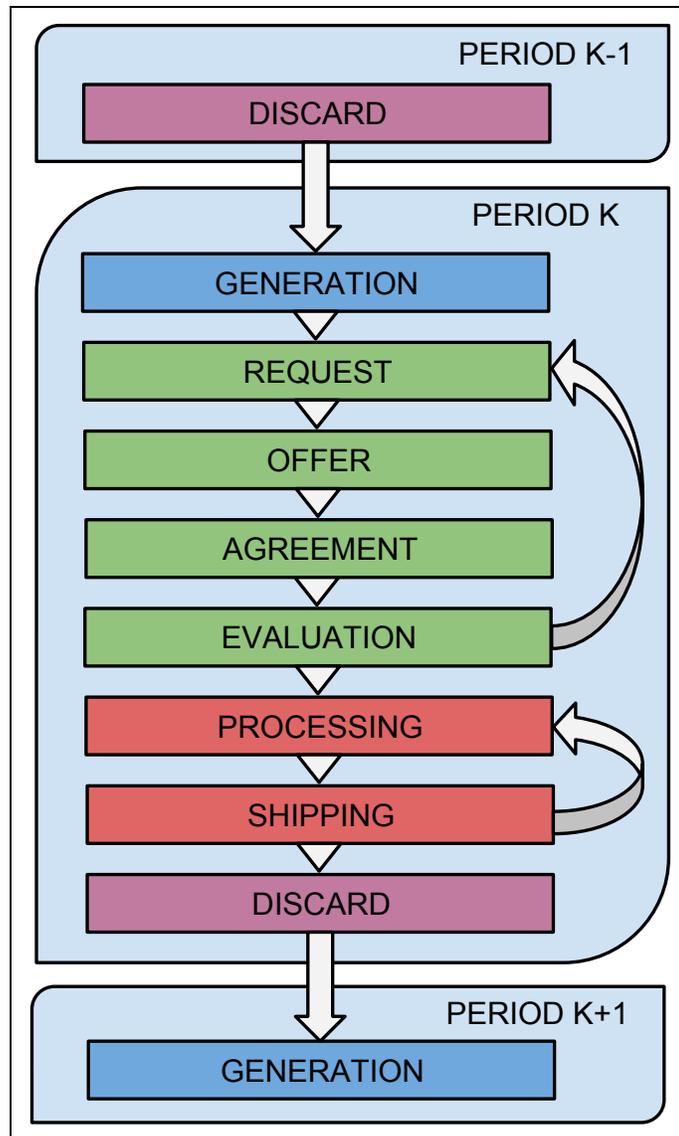
Figure 4: The phases of one period in the model.

phase, during which the actors evaluate the offers they have received, and choose to accept those offers that they deem suitable by sending agreement messages. After the agreement phase comes the evaluation phase, during which the actors evaluate the agreements they have, and have the option to break agreements that they cannot keep or deem unprofitable.

After the evaluation phase there are two options. If all the actors have a valid set of agreements, meaning that they can guarantee that they are able to process all the incoming commodities and ship away all the resulting products, the next phase is the processing phase. However, if some of the actors cannot guarantee this, the simulation returns to the request phase to allow those agents to attempt to fix the situation. This iteration can be repeated if needed. It is possible, in some cases,

that no solution can be found with some limited amount of iterations, which means that the execution is terminated to an error.

In the processing phase the actors use their processes to process commodities they have in their storage. The processing phase is followed by the shipping phase, during which the actors ship commodities according to their agreements. If after this phase all the commodities in the model have been transformed into end products and shipped to actors that can remove them from the simulation, the next phase is the discard phase, otherwise the processing phase comes again. If the supply chain is linear, this loop ends in finite amount of steps, as commodities are constantly being refined closer to end products.

The discard phase is the final phase of the period in this model. During the discard phase, end products such as deposited waste and recycled materials are removed from the actors that are able to discard them. After all of the end products have been removed from the actors, the next period begins.

4.7 Regulations

As one of the primary motivations for this kind of model is to model the effect of regulations to the waste management system, these regulations have to be somehow incorporated in the model. However, as these kinds of regulations can be very different in nature, they are difficult to be abstracted into objects. Thus, in this model regulations exist as a list of requirements that the behavior models of the actors have to fulfill. For example, landfills are required to pay tax for depositing waste, actors have a maximum amount of commodities that they can have in their storage at the end of the period, and agreements cannot be broken before the agreed duration has passed. Some of these regulations are parameterizable, for example, the amount of tax that has to be paid for certain actions.

4.8 Model Output

As this model is designed to replicate system-level behavior, the output of the model should be system-level data. However, as the events in the simulation happen at a lower level, some kind of aggregation has to be made in order to translate the individual events into large scale trends.

As the model generates various types of events, such as generation, shipping, processing and depositing of commodities, a multitude of different system-level behaviors can be observed. Examples of model outputs include distributions of different commodity values over time, commodity flows between different types of actors and amounts of different end products generated. Basically, any values that can be calculated from the individual events in the simulation can be used as an output of the model.

5 Waste Management of Finnish Daily Consumables Retail Shops

5.1 Overview

An actual waste management system was chosen as the system to be modeled. The system is the Finnish daily consumables retail shops and all the companies which take or could take part in the waste management of said shops. Only biowaste and biowaste related commodities were considered in this experimental model. The system was chosen as it is relatively well understood, and it was believed that sufficient data would be available to model the system.

However, using a real system as a reference presents some challenges. As statistical data from actual waste management systems is often inaccurate and the decision models of the various actors are not that well understood, many parts of the model are derived from expert opinions and different literature sources. The model is only approximate in many parts, but it is still sufficient for the needs of this project. For detailed reference on waste management systems in general, see for example [3] and [36].

The waste management system is composed of a few thousand shops, which generate together hundreds of thousands of tons of waste every year. The waste from the shops goes to landfills and waste treatment plants. The end products from the treatment plants go then to landfills, another treatment plant or are sold at the local or global market. In total, there are a few hundred different plants and landfills responsible for processing and depositing the waste.

5.2 Different Companies

The system is composed of various different companies, such as shops, transport companies, waste treatment plants and landfills. These companies can be roughly divided into four categories: those that generate, transport, process and deposit waste.

In this case the generators of waste are the shops, which come in many sizes from hypermarkets to small shops. The shops generate waste as a byproduct of selling consumables to consumers. The waste generated includes, for example, expired food items, packaging materials and mixed waste from waste bins. The amount of waste generated correlates with the turnover of the shop, which in turn correlates with the size and location of the shop, and also with the time of the year. The generated waste can also be sorted in different ways, which are usually correlated with the chain and the region of the shop.

Transport companies are the transporters. They come in various sizes, from large chains to independent operators. The transport companies transport waste from one place to another, but the party who decides where the waste is transported to may vary. For example, a shop may order a transport service for some type of waste from an independent operator, specifying where to take the waste. In this case, the shop is responsible for paying the transport operator for transport and

the receiver for acceptance of the waste. On the other hand, the shop may buy the waste transport service from a large company that decides the destination based on its own agreements with different treatment plants. In this case, the shop pays a service fee, which depends on the amount and type of waste picked up. The costs incurred to a transport company for transporting waste correlate with the distance and amount of waste transferred.

Treatment plants are the processors. For example, composting plants, waste incineration plants and biogas plants are all waste processors. A waste processor is generally able to process only certain types of commodities. The processor charges a gate fee, usually euros per ton of waste, for receiving commodities. The gate fee may vary based on the trading partner, type of waste, and the exact composition of the batch of waste, such as dampness or amount of impurities. The processor companies use processes to transform the incoming commodities into other types of commodities, which are usually more valuable than the incoming commodities. However, the company has to incur the expenses of having and using the process, which correlate with the type and capacity of the process and the amount of materials processed.

In this model, there are two ways for commodities to exit the system. Either they are deposited to a landfill or sold at a market. In this sense, both landfills and markets can be classified as depositors. Landfills accept different types of waste, and charge a fixed gate fee that depends on the landfill and the type of the waste. Landfills may have to pay a tax for depositing some types of waste, which is added to the gate fee. Markets, on the other hand, are different kind of entities, as they are formed of various different companies that deal in the relevant commodity. The price at which some commodity can be sold at a market depends on a large amount of variables, such as the local and global supply and demand.

5.3   Market Structure

The waste management system is composed of various companies that offer services to each other. As the companies are situated in different parts of the country, some companies are closer to each other than others. Shorter distances usually translate into lower transportation fees and thus lower prices. Therefore, the companies are usually limited in dealing with the agents near themselves, usually under a few hundred kilometers away from the company's location.

The companies can be assumed to have a good understanding of the companies in their local market that offer services they are interested in acquiring. For example, a shop should know what kinds of waste companies operate in its area and what kinds of waste fractions each of them will accept. Thus each company can be assumed to be able to contact each other and ask the price of some service the other can provide.

In reality, negotiation of contracts can be quite a complex process, possibly involving many parties over a long period of time, or conversely a very simple one, if the service can be simply ordered based on a known price. The contracts can also vary in their type, length, amount of details and amount of parties involved.

5.4   Environment

The companies can be seen to exist in many kinds of environments. From the perspective of this modeling project, the most important ones are the physical, economic and regulative environments.

The physical environment includes the physical and technical limitations that the companies face, and the nature. Physical limitations include, for example, the topology of the world, meaning that the companies are at different distances from each other. Technical limitations include, for example, the types of commodities some process can produce given certain input commodities. The nature means the natural world, such as the atmosphere and soil, which the companies' actions can affect.

The economic environment means the financial aspects and limitations the companies face. For example, if a company's expenses are constantly greater than its incomes, at some point the company will become insolvent and be bankrupt.

The regulative environment refers to all the laws and regulations the company has to take into account in its actions. For example, taxes that have to be paid for certain actions are part of the regulative environment.

5.5   Supporting Data

Data for the model was collected and combined from many different sources. For example, the amount of different types of stores in different regions was calculated by multiplying the total number of stores in the region, taken from [37], with the share of the shop type in that region, taken from [38]. Price and cost references were taken, for example, from [39] and [40]. Locations of different processing plants were taken, for example, from relevant company websites.

As the data was collected from many different sources, all of it may not be from the same year or region. In general, the data was chosen or approximated to represent the status of the waste management industry in Finland in the year 2008. Data for some of the parameters in the model was not found from any public references, so it was approximated based on the general knowledge of the waste management researchers working on the project.

An example of the different types of data that was collected is shown in table 1.

Region	Number of shops	Est. total sales (Meur)	Est. biowaste generation (tons/a)
Uusimaa	740	3699	8559
Varsinais-Suomi	371	1203	2912
Itä-Uusimaa	71	230	557
Satakunta	190	622	1505
Kanta-Häme	136	448	1084
Pirkanmaa	366	1225	2965
Päijät-Häme	136	531	1285
Kymenlaakso	130	496	1200
Etelä-Karjala	95	367	888
Etelä-Savo	143	426	1031
Pohjois-Savo	193	637	1542
Pohjois-Karjala	139	426	1031
Keski-Suomi	204	699	1692
Etelä-Pohjanmaa	175	505	1222
Pohjanmaa	160	439	1063
Keski-Pohjanmaa	63	188	455
Pohjois-Pohjanmaa	296	948	2294
Kainuu	76	227	549
Total	3684	13934	31835

Table 1: Number of retail shops, estimated total sales and estimated biowaste generation by region in Finland in 2008. Combined from different sources [37, 38, 41, 42].

6 Model Construction

6.1 Overview

In order to be able to analyze the suitability of ABMS for implementing WVCA models, an experimental model was constructed. The system to be modeled was the waste management of the Finnish daily consumables retail shops, described in section 5.

The model construction was done in two phases. In the first phase, a simple model was constructed to explore the capabilities of the modeling methodology. After the first phase was complete, a more accurate model was designed and implemented. However, it was noticed during the project that there actually does not exist enough data for the model to be properly validated. For this reason, the project workflow deviated somewhat from the planned.

In this section, the experiences from the modeling project are described, along with details of the implementation and explanation of the various design choices made.

6.2 Project Workflow

The modeling project followed approximately the general modeling workflow presented in section 2. However, the project does not include proper model validation, which led into changes in the experimentation, analysis and presentation phases. The workflow is described next as it progressed during the project.

The planning phase started at mid-January 2012, when the project team, including two waste management researchers and the author, met a few times and discussed the project. In the project, the waste management researchers were responsible for designing the WVCA model and collecting the data for the model, and the author was responsible for the technical implementation using the ABMS modeling paradigm. In these meetings the outline and goals of the project were set, as well as the general outline of the system to be modeled and the WVCA model. After this, the author started to familiarize himself with the modeling method and to evaluate the different modeling tools that could be used for implementing the model.

The first definition phase was a meeting with the project team at the end of January, where an outline of the first model and the tools to be used were decided. The outline was quite flexible at this point, as the actual capabilities of the modeling method were not yet well understood.

The first implementation phase started at the beginning of February, and ended at the end of the month. During the implementation phase, the project team kept in touch using email to communicate the details of the implementation, as the team worked at two different locations and there were not so many details that needed to be discussed concerning the implementation.

After the first version of the simulation software was completed, it was evaluated in a team meeting. Based on the evaluation, the second definition phase was started

to create a more accurate definition for the second version of the model. This definition phase lasted until the end of March.

The second implementation phase started at the end of March, and lasted until the beginning of June. At the beginning of the phase, the project team kept again contact using email, but near the end of the phase the team worked some days of the week at the same location, as some aspects of the model needed more interaction within the project team in order to be implemented correctly.

A partial validation of the model was done after the second implementation phase, in the beginning of June. The validation consisted of examining three example cases and estimating the feasibility of the system-wide trends based on the researchers' knowledge of the system. The model was calibrated manually during the validation. The resulting model yielded results that were reasonably believable. However, no objective validation was done, as it would have required more time and resources to be allocated to this project for two reasons. First, as proper statistical data concerning many of the properties of the waste management system, such as commodity unit prices, was not readily available at the time of model construction, that data would have had to be collected before this kind of validation could take place. Collecting this data would have required a considerable amount of work. Second, as the model is quite complex with hundreds of parameters, properly calibrating the model would also have required a considerable amount of work. As the scope of the project was limited both by funding and the scope and time limitations of a master's thesis, proper validation was not possible.

As the model validity could not be guaranteed, no actual experiments were run on the model. Thus, the scope of the experimentation phase was changed from running experiments on the model to obtaining feedback from a group of experts. Two sessions were organized during June to obtain feedback on the model feasibility. These sessions are described in section 7.

The scope of the analysis phase was changed from analyzing results to analyzing the feasibility of the ABMS paradigm for implementing WVCA models. The analysis phase started in the beginning of July. The analysis phase consists of writing this thesis and discussing the future of the model together with the project group.

The presentation phase started in August, when writing of this thesis was almost finished. An article was written concerning the WVCA model [43], and the results of the project were presented to the project sponsors. The results of this study are also used in the further development of the WVCA model.

6.3   Model Definition

The model definition was constructed gradually as the project progressed. The initial definition was a rough sketch which acted as a guideline for the first implementation phase. The definition was improved and made more specific based on experiences from the first implementation phase. However, as questions concerning specific aspects of the model arose during the second implementation phase, the definition was specified respectively. In this subsection the most important aspects of the final model definition are described.

The model structure was created specifically for this application, but inspiration was taken from agent-based supply chain models, such as [44] and [45].

Scope of the Model

The model was limited to contain all the shops from all the regions of Finland, except from Lappi and Ahvenanmaa. Neither of these regions contains a considerable amount of shops or waste management industry, and their landscape is different from the other regions. Lappi is characterized by very long transport distances and Ahvenanmaa is composed mainly of archipelago. Also, the waste management in these regions is largely self-contained, meaning that they do not have a large effect on the waste management in other regions. The remaining 17 regions were seen as a sufficiently complex system for the purposes of the project.

The model was also limited to include only the biowaste sector, as this limited the amount of data that needed to be collected for parameterizing the model. This means, that only companies that accept biowaste related commodities were included in the model. This limited the types of processors to biogas, composting, incineration and mixed waste preprocessing plants, and types of commodities to clean, packed and mixed biowaste, compost product, biowaste reject, ash, biogas, electricity and district heat.

One period in the model corresponds to one quarter (three months) in real time. This duration was deemed sufficient to represent the system-wide dynamics, and it is also compatible with the fiscal quarters that companies commonly use in accounting. The model is designed for simulations lasting from a few years to a few decades.

To simplify the transportation in the model, the transport companies were left out of the model. This means that it was supposed that transportation is a generic service available to all the agents, and it costs a predefined amount that depends on the type of waste transported and the transport distance. Furthermore, it was assumed that the agent supplying the waste is the one responsible for transporting the waste to the receiver, and thus, the supplier has to pay all the costs of transportation.

Agents

The agents are divided into shop agents, processor agents, landfills and markets. The processor agents are further divided into subtypes, based on the process the agent has. The properties of the different agent types are elaborated next.

The shops were divided into six different categories based on their size. The classes are from largest to smallest: hypermarket, large supermarket, small supermarket, large market, small market and small shop. Each shop belongs to a chain, but as the data collected does not differentiate between chains, all the shops belong to an undefined chain.

Each shop belongs to a certain region, which defines the location of the shop. The amount of different types of shops to be generated to each region was approximated from statistics. As there is a large amount of shops, the locations were taken randomly from a 2D Gaussian distribution, which is centered on the main city of

the region. The regions are illustrated in figure 5. The location is required to be inside the borders of the region, approximated by a polygon. An example of the distribution of shops in the map is shown in figure 7.
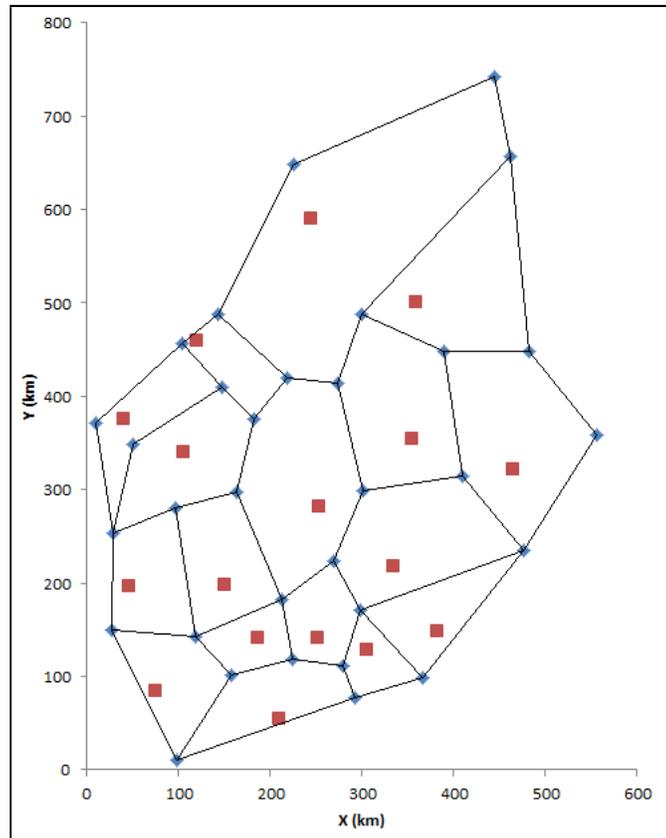

Figure 5: The approximation of the regions of Finland used in the model. Blue diamonds represent the locations of the corners of the polygons that define the regions, black lines represent the edges of the polygons, and red squares represent the locations of the central cities of the regions. For reference, a map with the actual regions is shown in figure 6.

The amount of waste generated by a shop in each period depends linearly of the turnover of the shop. The turnover, on the other hand, depends on the type of the shop, the region the shop belongs to and the time of the year (i.e. current quarter). A lookup table for the turnovers was made using statistical data.

The distribution of the total amount of waste generated depends on the type of the shop and the chain the shop belongs to. For example, if on average 2% of the total amount of waste generated by a hypermarket that belongs to chain A is packed biowaste, 7% clean biowaste and 5% mixed biowaste, and if the total amount of waste generated in this period is X, then the amounts of packed, clean and mixed biowaste generated are 0.02 X, 0.07 X and 0.05 X respectively. A lookup table of the distributions was made based on statistical data.

The goal of the shop agents is to make agreements that guarantee that the shop
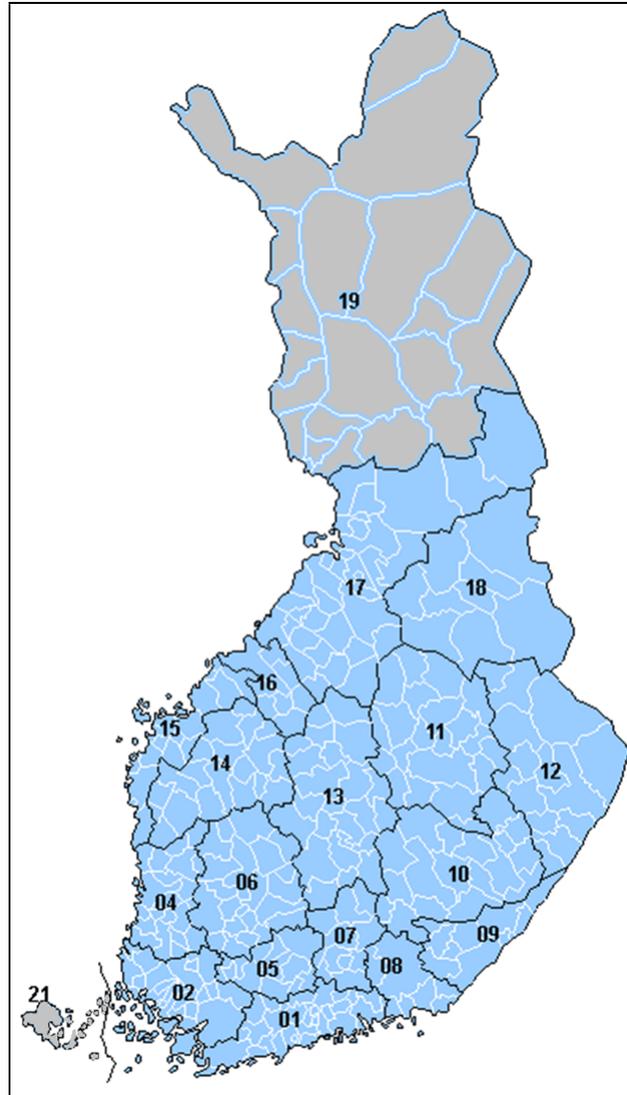
Figure 6: The map of Finland with region borders shown, modified from [2]. The grey areas are the Lappi and Ahvenanmaa regions, which are not taken into account in this model.

gets rid of all the waste that it generates every period. In order to get offers, the shop sends requests to all the other agents in its neighborhood that can accept the types of waste the shop generates, unless the shop already has an agreement that cannot be broken at this time. If the shop gets multiple competing offers for a certain type of waste, it chooses the cheapest one. However, if the shop already has an agreement that could be broken for that type of waste, the new offer needs to be a certain amount cheaper than the existing one in order to be accepted.

The processor agents are classified based on the process they have. The processes included biogas, composting, incineration and mixed waste preprocessing plants. The types of commodities that the different types of processor agents can receive and
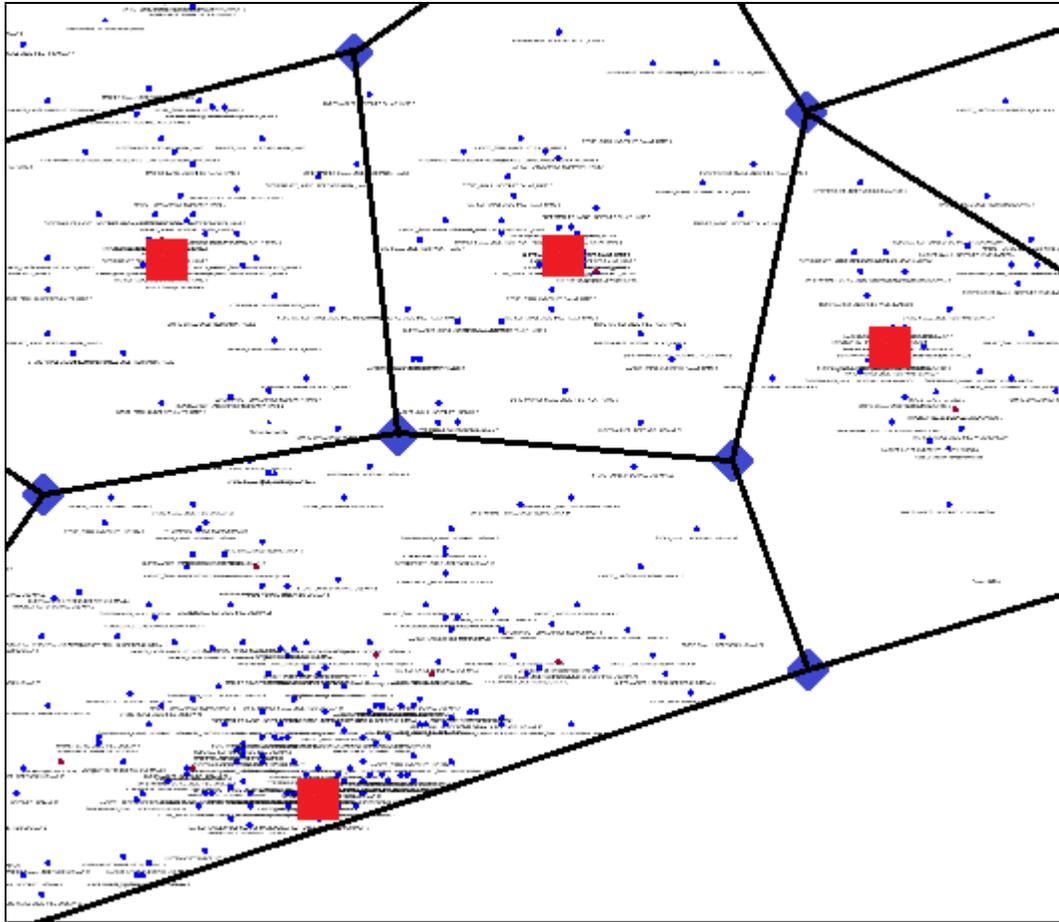
Figure 7: An example of the distribution of agents in the southern part of Finland (Uusimaa, Kanta-Häme, Päijät-Häme and Kymenlaakso regions). The approximate borders of the regions and locations of central cities are marked on top of an image taken from the simulation visualization, showing agent locations with names. A large majority of the agents shown are shops.

produce are illustrated in figure 8. The amount of different outputs depends linearly on the amount of input, with different coefficients for different types of outputs. For example, a composting plant could produce 0.3 tons of compost product and 0.02 tons of biowaste reject per one ton of clean biowaste. The process has a fixed upkeep cost based on the process type and a linear processing cost based on the type and amount of commodity processed. The process also has a maximum capacity for commodities that can be processed per period, and the types of commodities that can be processed in a process can be limited by setting the first and the last period that each type of commodity can be processed.

The processors make offers based on the expected value of the processed end products, taking into account the expected processing and transport costs and the minimum profit margin. The aim of the processors is to generate as much profit as possible. To simulate the market behavior, if a processor agent gets a request
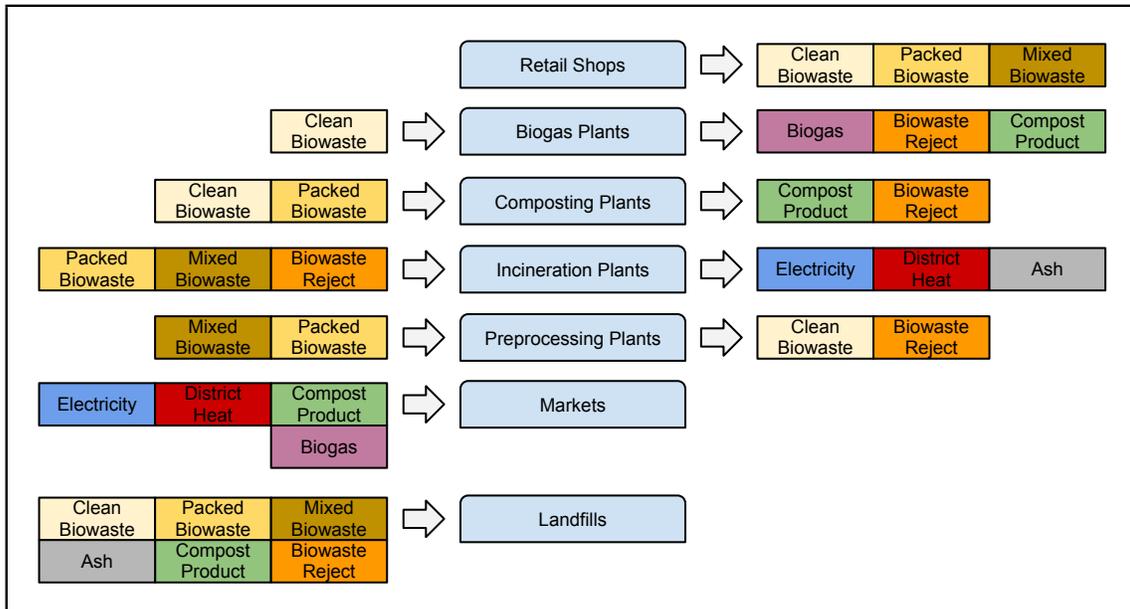
Figure 8: Input and output commodity types for each agent type. Commodities the agent type can receive are on the left and commodities the agent type can produce are on the right.

from another agent that has accepted the previous offer it sent, it tries to drive the price up by increasing the fee a small amount. Conversely, if the previous agreement was not accepted, it tries to undercut the prices of its competitors by lowering the fee by a small amount. However, the agent will never make offers that would be clearly unprofitable, which sets a limit to the lowest price it can offer. A pseudo-code example of an algorithm used to send offers in response to requests is shown in figure 9. The processor agents use similar logic with the shop agents for getting agreements for outgoing commodities. An example of an agreement network that is formed during the simulation is shown in figure 10.

```
send offers {
  for each unanswered request {
    base value = the estimated unit value of commodities
                 received from processing requested type
                 minus processing and transport costs
    last offer = last similar offer sent to this agent
    if there is no last similar offer {
      offer price = base value plus initial margin
    }
    else {
      if the last similar offer was accepted {
        offer price = price of previous offer increased
                      by a small amount
      }
      else {
        offer price = price of previous offer decreased
                      by a small amount
      }
      if the offer price is lower than the base value
      plus minimum margin {
        offer price = base value plus minimum margin
      }
    }
    if current allocated capacity plus the size of this
    offer is larger than maximum capacity {
      offer = offer that is impossible to accept
    }
    else {
      offer = offer to accept requested amount of
              requested commodity type for the default
              duration of agreements
    }
    send offer and mark request as answered
  }
}
```

Figure 9: A simplified pseudo-code version of the algorithm used by the processor agents to respond to requests.

The depositors are composed of landfills and markets. All the commodities that these agents receive are considered end products from the viewpoint of transactions in the system and will be removed from the simulation at the end of every period.
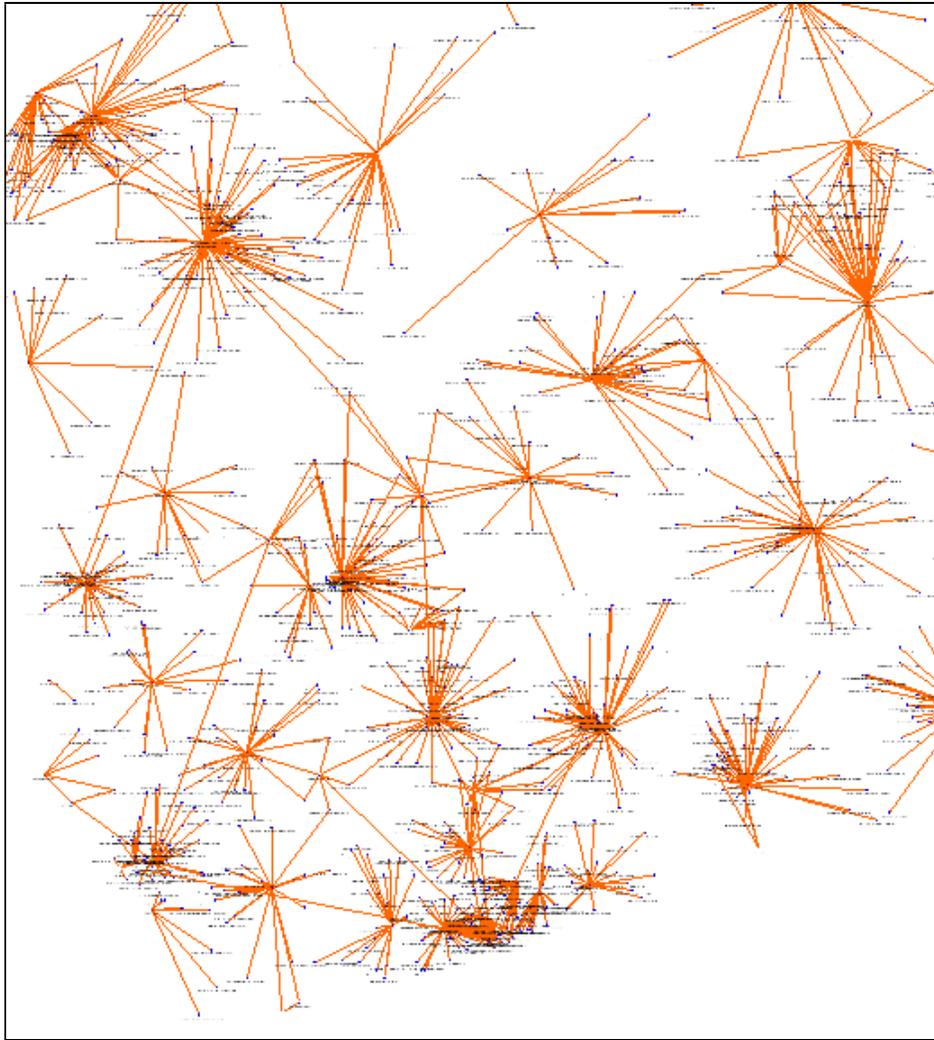
Figure 10: An example of the agreement network that is formed after negotiations, orange lines represent agreements between agents.

The commodities that landfills and markets can receive are illustrated in figure 8.

Landfills have a list of commodities that they can accept, and the types can be limited by setting the first and the last period that each type can be accepted. Each type of commodity has also a fixed base gate fee, which can be given separately for each landfill. The landfill has to pay tax for depositing waste, which is given separately for every period in euros per ton. As the landfill agent does not supply anything, it does not send requests. The landfill agents always offer the base gate fee added with the tax when sending offers.

Markets have a list of commodities that they accept, and each commodity has a set market price for each period. The market agent does not send requests, as it does not supply anything, and it always offers the market price when sending offers.

Environment and Simulation Functions

In addition to the agents, the simulation contains other entities, such as the scheduler, the topography of the world, the mail service, the transport service, the economy service, the data collector, the logger, the parser and the initializer. These are the environment and simulation function entities.

The scheduler is responsible for timing the agents' actions so that they are able to be simulated in a sequential order. The agents have an interface for allowing different types of behaviors, such as sending requests or processing commodities. The scheduler has also interfaces to the mail service, transport service and the market economy to allow also them to act in turn.

The topography of the world is a 2D Euclidean space, where all the agents are situated. The transport distance between the agents is approximated to be the shortest distance between the agents' locations. All the agents are in the same context, but they are only allowed to communicate with other agents inside their neighborhood, which is defined by a maximum distance to a neighbor (e.g. 100 km).

The mail service and the transport service are entities that the agents use to send messages and shipments to each other. Both of these services are synchronized, meaning that the messages and shipments are delivered to the recipient only after all of the agents have had the opportunity to send messages or shipments respectively.

The economy service observes the agents and checks whether they act profitably or not. If an agent is unprofitable for too many consecutive periods, it is removed from the simulation. A future feature of this entity will be the ability to create new agents to free sites if the area seems suitable for a new company.

The data collector is responsible for receiving event messages from the agents. The agents are required to send event messages when certain event happen, such as sending or processing commodities. These event messages are stored during the period, and processed at the end of it. The data collector calculates different statistical figures from the events, and allows the agents to ask these figures from the past periods. For example, the agents could ask the average price of a certain commodity in the past. The data collector also calculates different system-wide trends from the data, and uses the logger to write them into an output file.

The logger is responsible for writing logs of the simulation, as well as the simulation output file. All the other entities can use the logger for writing arbitrary data to the simulation log file, which can be used to trace the progression of the simulation if needed.

The initializer and the parser are responsible for initializing the simulation. The initializer uses the parser to read the simulation parameter file during the initialization. The parser reads the parameters from predetermined locations in the file and sets the parameters of all the other simulation entities to correct values. The initializer creates and initializes the agents and other simulation entities.

6.4 Modeling Tools

As the implementation is time-consuming, it is important to have adequate tools to aid the implementation. In this project the Repast Simphony 2.0 beta toolkit [46] was used for the implementation of the model. The other toolkits that were considered were MASON [47], NetLogo [48] and Swarm [49]. A review of agent based modeling platforms by Nikolai and Madei [50] was used for identifying prospective toolkits, and a shortlist was made based on further analysis of the platforms.

MASON is a light execution and visualization environment for agent based models, but needs a separate Java programming environment to be used for software development. As the package is very light, it does not include many tools to help the modeling effort.

NetLogo is an entire modeling environment for multi-agent simulations. NetLogo includes an integrated programming, execution and visualization environment, and is quite easy to use. However, the programming environment is quite limited, as all the programming has to be done using the NetLogo programming language, and the entire model has to be implemented in one text file.

Swarm is a development environment based on the Swarm framework for agent based models. It is one of the oldest modeling tools for agent based models, and does not include a programming or visualization environment. Also, it is developed to be run in UNIX operating systems, which causes some additional work when using Windows.

Repast Simphony is one of the newest tools for agent based modeling. The development environment is built on the Eclipse Java programming environment [51], and includes also an integrated execution and visualization environment that can be integrated to different analysis tools, if available. Repast Simphony was chosen because it provides all the tools needed for development of agent based models. A screen capture of the Eclipse programming environment is shown in figure 11 and a screen capture of the Repast Simphony execution and visualization environment is shown in figure 12.

Also, as the model had a large amount of parameters and generated a large amount of data when executed, Excel was used for both handling the parameter values and the simulation results, as it was the preferred by the researchers that would be using the model.
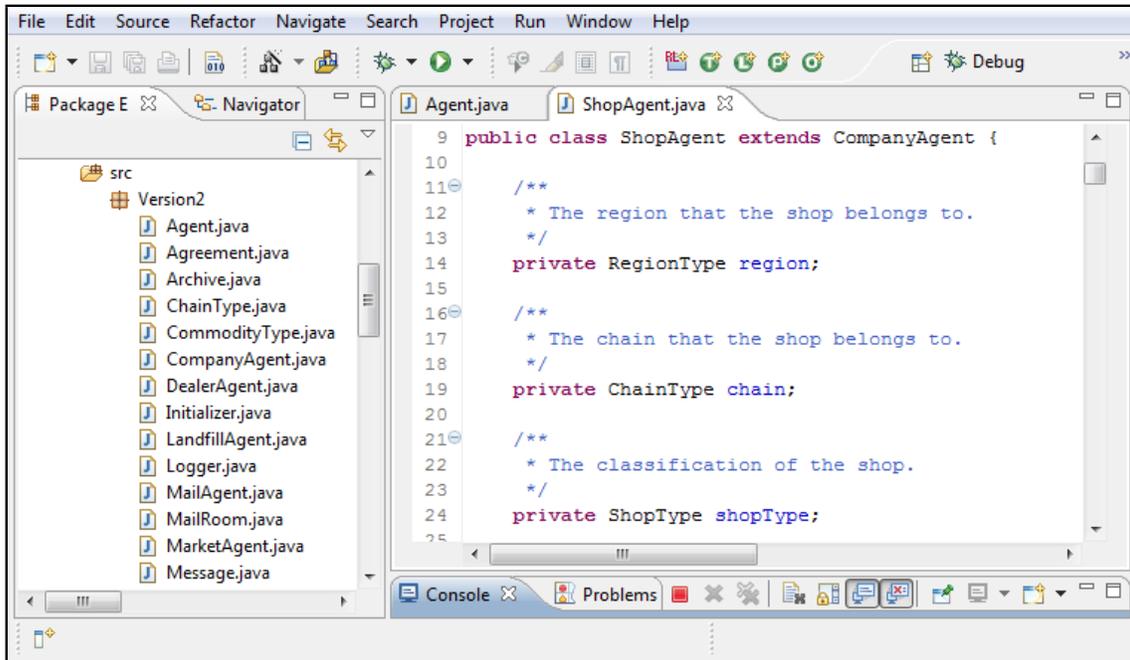
Figure 11: A screen capture of the Eclipse programming environment.
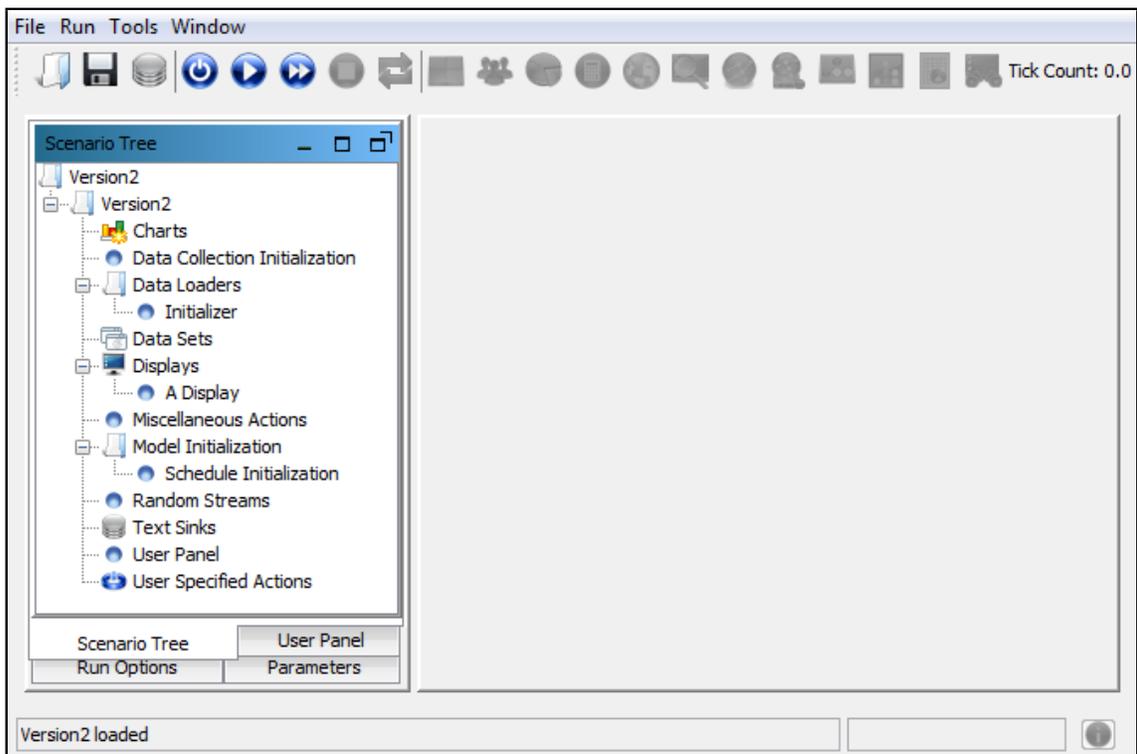

Figure 12: A screen capture of the Repast Simphony execution and visualization environment.

6.5   Model Implementation

The implementation of the model consists of transforming the requirements of the model into program code and verifying the implementation. In this subsection the implementation of the second version of the model is described with comments.

The model was implemented using the object-oriented Java programming language. As agents are quite close to the objects in object-oriented programming, Java is a natural choice for the implementation language. Other object-oriented languages, such as C++, C# or Python, could have been used as well, if the toolkit would have supported them.

The implementation was planned by making a UML (Unified Modeling Language) class diagram of the program, which described the different classes, their methods and relation to each other. The structure of the classes was programmed based on the UML diagram and the specific algorithms were programmed based on the model requirements. The program implementation contains some 6000 lines of Java code (including comments and blank lines), and is separated into 24 different classes. An excerpt from the finished UML diagram can be seen in figure 13 and a simplification of the full diagram is illustrated in figure 14.

The different features of the model were tested manually to verify their functionality. Unit testing was considered, but was postponed for two reasons. First, the tests would have had to be rewritten many times, as the definitions of many of the features were subject to change during the implementation. Second, the manual testing appeared to result in sufficiently bug-free behavior for the time being. Also VOMAS [35] was considered, but postponed for similar reasons.
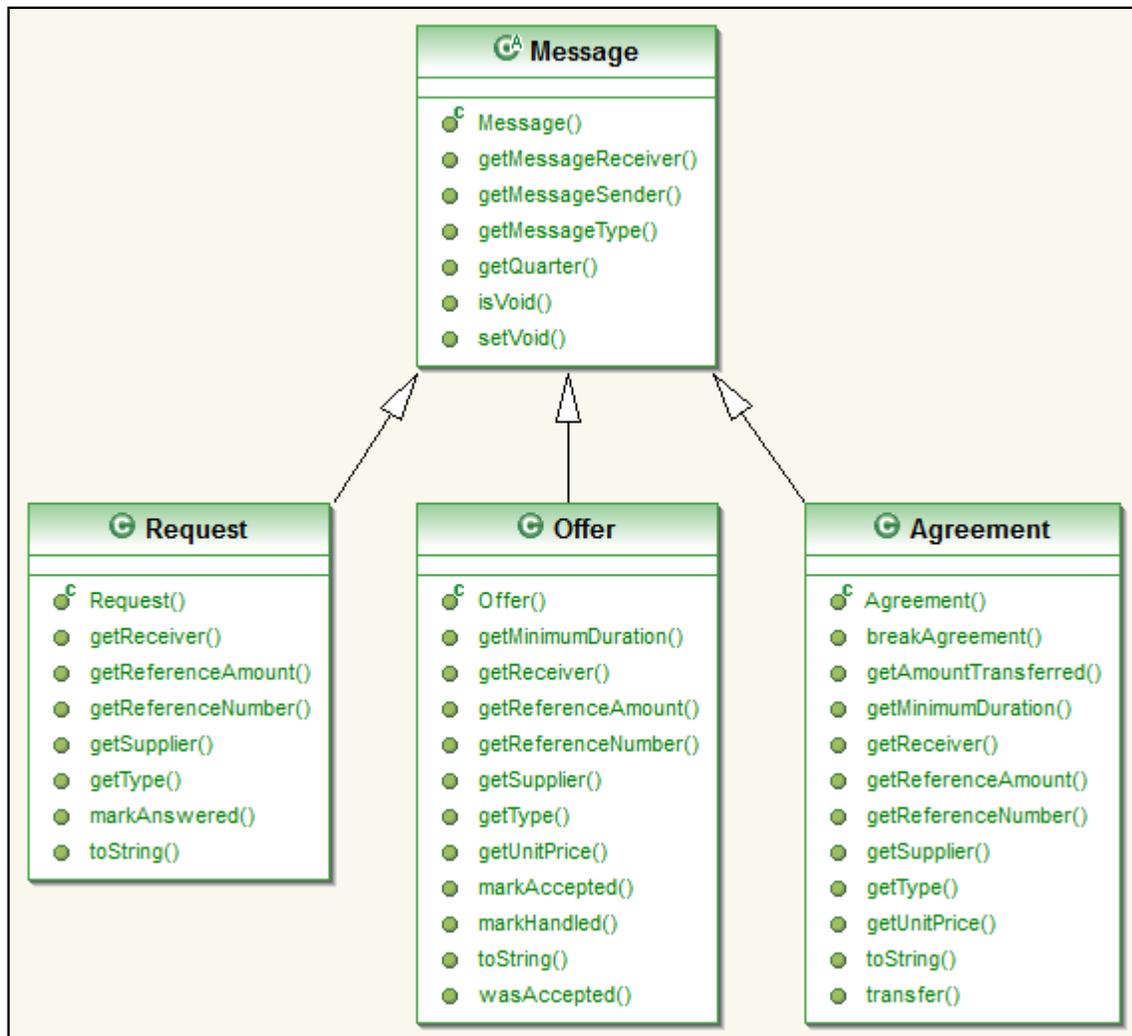
Figure 13: An excerpt of the finished UML class diagram showing the Request, Offer and Agreement classes that are inherited from the abstract Message class.
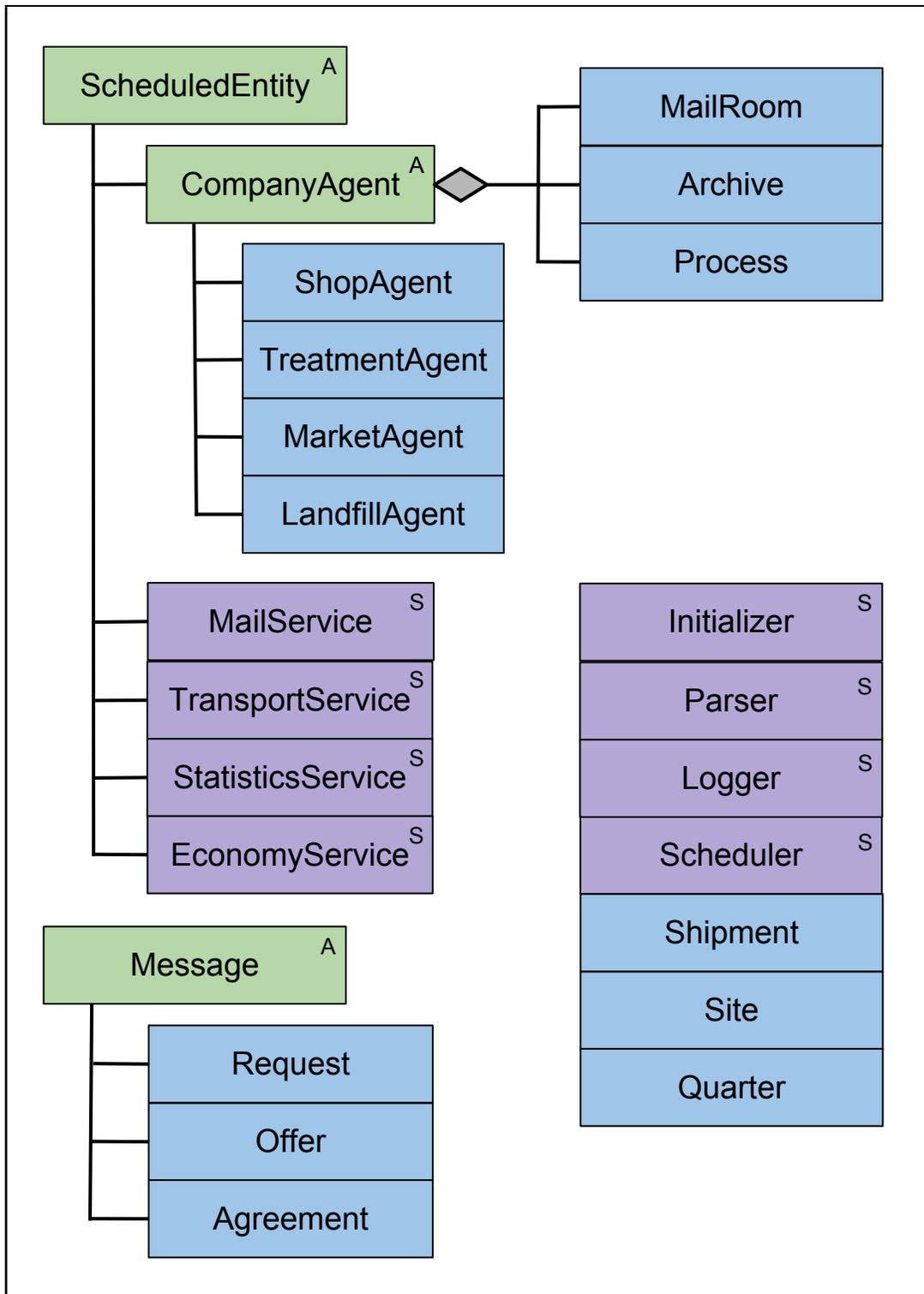
Figure 14: A simplification of the finished UML class diagram showing all the classes in the model. 'A' refers to an abstract class that cannot be instantiated. 'S' refers to a static class, meaning that only one instance of it exists.

6.6 Model Interfaces

The model is run using a runtime environment that is included in the Repast toolkit. The runtime environment contains buttons for loading the model and running it either continuously or step-by-step. The environment also contains a configurable visualization environment, which can be used to observe some aspects of the model during runtime. An example of the Repast environment is shown in figure 12 and examples of the visualization are shown in figures 7 and 10.

The model parameters are stored in a spreadsheet file, which contains 16 sheets for different parameters. There are some 1500 parameters describing the structure of some 1200 agents and the environment, and some 20 parameters for the behavior models. An excerpt from the file is shown in figure 15.

Selected data is collected from the model during the simulation, and trends are calculated and printed in a spreadsheet output file after the simulation. Examples of different types of trend data that can be exported from the simulation are shown in shown in figures 16-27.

Shop chain	UNDEFINED	-	-
Shop type	HYPERMARKET	SUPERMARKET_LARGE	SUPERMARKET_SMALL
Region name	Amount of shops to be generated		
UUSIMAA	31	122	126
VARSINAISSUOMI	16	63	46
SATAKUNTA	9	23	32
KANTA_HAME	8	28	20
PIRKANMAA	16	62	46
PAIJAT_HAME	8	28	20
KYMENLAAKSO	6	20	25
ETELA_KARJALA	5	10	15
ETELA_SAVO	10	26	5
POHJOIS_SAVO	10	23	23
POHJOIS_KARJALA	7	17	17
KESKISUOMI	10	46	19
ETELA_POHJANMAA	13	6	19
POHJANMAA	12	6	17
KESKI_POHJANMAA	5	2	7
POHJOIS_POHJANMAA	15	26	65
KAINUU	4	7	17

Figure 15: An excerpt of the parameter file, showing the amounts of some of the different types of shops that are generated into the regions.

6.7   Simulation Results

To demonstrate the capabilities of the model, three different scenarios concerning landfilling regulations were constructed and simulated for the duration of three years. Relevant trend data was calculated from the simulation results to demonstrate the effects of different regulations. It should be noted that these simulation results are based on a model that has not been validated, so the behavior may not correspond with the actual system. However, they indicate that reasonably plausible results can be generated already at this stage.

The first scenario was a static situation, and acts as a reference for the other scenarios. In this scenario landfilling of packed biowaste, clean biowaste, mixed biowaste, biowaste reject, ash and compost product was allowed, but landfills were required to pay a tax of 30 euros per ton for depositing any type of waste. The initial landfill gate fee was on average 11 euros per ton, so the actual gate fee was on average 41 euros per ton. Market prices and other parameters that usually vary every quarter were kept static during the simulation to minimize their effects on the behavior. The second scenario was otherwise identical to the first one, except that the landfilling tax started from 30 euros per ton and increased by 5 euros per ton after every quarter. The third scenario was identical to the first one, except that the gate fees of the landfills were set to 1000 euros per ton for all waste types, except for ash and biowaste reject. This practically banned landfilling unless landfilling was the only choice.

The first trend is the amount of commodities transported in the model, measured in ton kilometers. Electricity, district heat and biogas are not usually transported by using vehicles, so they were left out from these trends. These trends are shown in figures 16, 17 and 18. They could be used, for example, to calculate the environmental effects of transporting waste.

The second trend is the value of commodities from different plants. The values are based on the agreements the agents have with each other. The values are taken at the time the receiver receives the shipment. Negative values refer to the fact, that the supplier had to pay the receiver for accepting the waste. If a plant did not receive any amount of some type of commodity, that trend line is not shown. These trends are shown in figures 19, 20 and 21.

The third trend is the amount of commodities deposited to landfills. These trends are shown in figures 22, 23 and 24. They could be used, for example, to calculate the environmental effects of depositing waste.

The fourth trend is the amount of commodities sold to the market. These trends are shown in figures 25, 26 and 27.

Various behaviors can be observed from these trend figures. First of all, transported ton-kilometers seem to increase for all transportable commodities as landfilling becomes less attractive. This can be seen by comparing figures 16, 17 and 18. This could result from other options that are further away becoming more attractive, which results in longer transportation routes. Of course, commodities that can only be deposited to landfills in this model, such as ash, are most likely not transported further away, but are more likely transported in larger quantities as, for

example, incineration plants start to become more attractive trading partners and thus produce more of it.

As the price of landfilling goes up, it appears to slightly drag the general market prices up as well. This can be seen in figure 20. This could result from the dynamic situation where new markets open up little by little for treatment agents, as landfills are constantly becoming more and more expensive. This may give some agents possibilities to get even more expensive offers accepted, as the market situation is not stable. This may also partly result from the increasing expenses from disposing of process side products adding up to the prices. However, if landfilling becomes prohibitively expensive, the prices seem to behave very similarly as in the situation when landfilling was not much regulated. This could result from the competitive but stable market situation, which keeps the prices low. This may also partly result from the costs of disposing process side products becoming low again. It can be also noticed that preprocessing plants in the model have become economically viable only in the third scenario. This is to be expected, as they transform one type of waste into another, which generally does not produce that much value.

The amount of commodities deposited to landfills seems to correlate negatively with the cost of landfilling. This can be seen from figures 22, 23 and 24. This is to be expected, as the agents prefer options that are more economical to themselves. It can be noted that a small amount of different commodities are still landfilled apart from ash and biowaste reject. This is because in the simulation there are a few agents that have been generated to remote locations, where there are no treatment agents in their neighborhood. To allow the simulation to execute normally, these agents must be allowed to landfill their wastes, even though at a great cost.

The amount of commodities sold at the market seems to correlate positively with the cost of landfilling. This can be seen from figures 25, 26 and 27. This is also to be expected, as when treatment agents become more attractive trading partners, they also process more commodities, which results in a larger amount of commodities to be sold to the market.

Additionally, in figure 25 it can be noticed that the amount of biogas produced increases during the third year. This likely results from the fact, that at the same time as this happens, the average price of reception in biogas plants becomes lower than the average price of landfilling, which can be seen in figure 19.

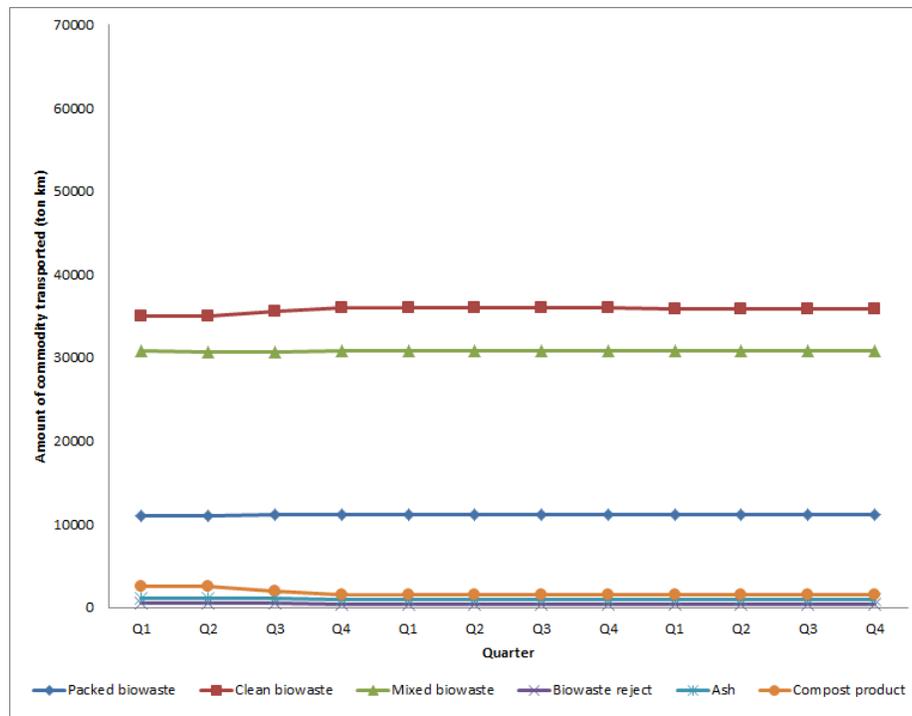In general, the model seems to be able to produce behavior that could be expected from the actual system.

Figure 16: Transported commodities in ton kilometers per quarter in scenario 1.
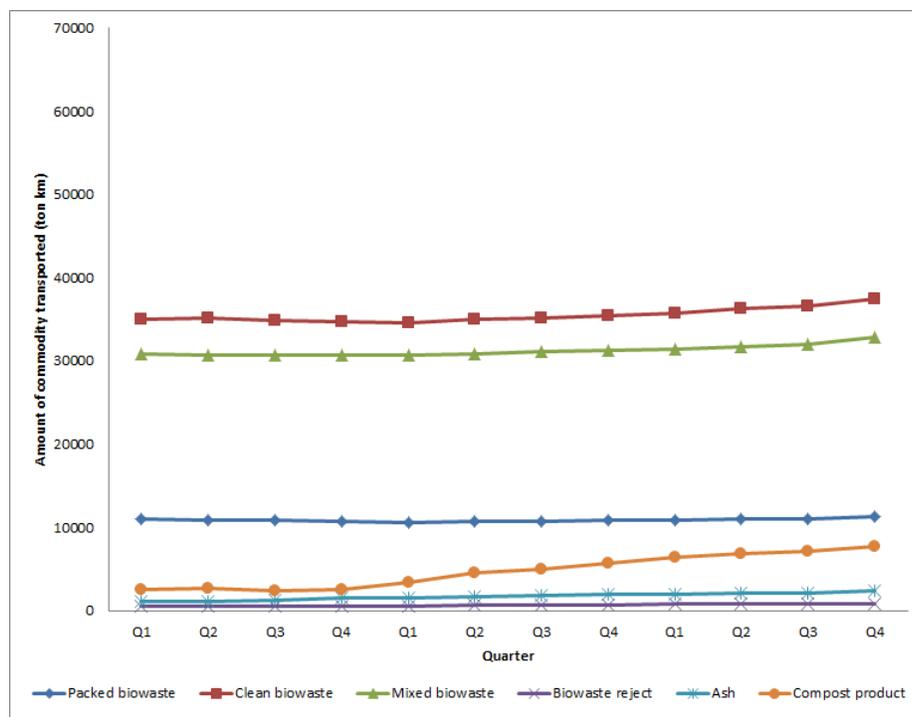

Figure 17: Transported commodities in ton kilometers per quarter in scenario 2.
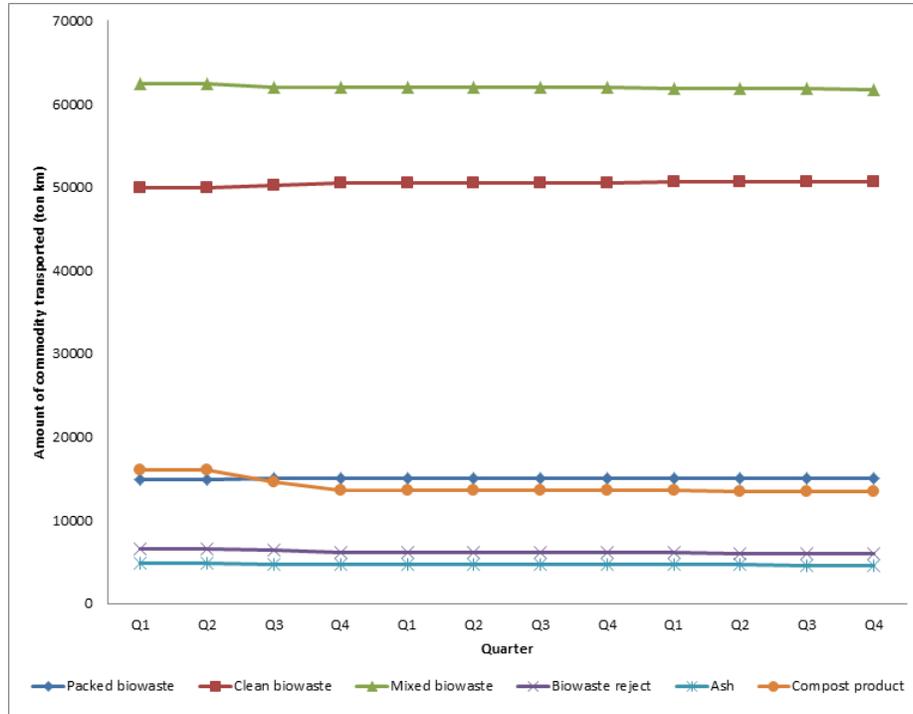
Figure 18: Transported commodities in ton kilometers per quarter in scenario 3.
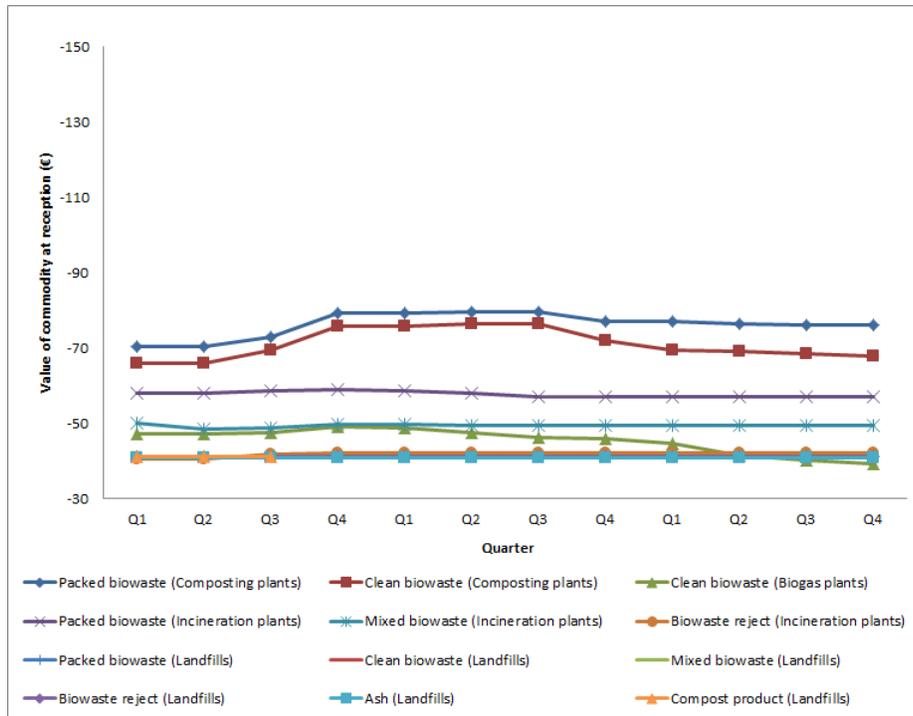
Figure 19: Average value of commodities at reception in euros per ton from different types of sites per quarter in scenario 1. All landfill gate fees overlap each other.
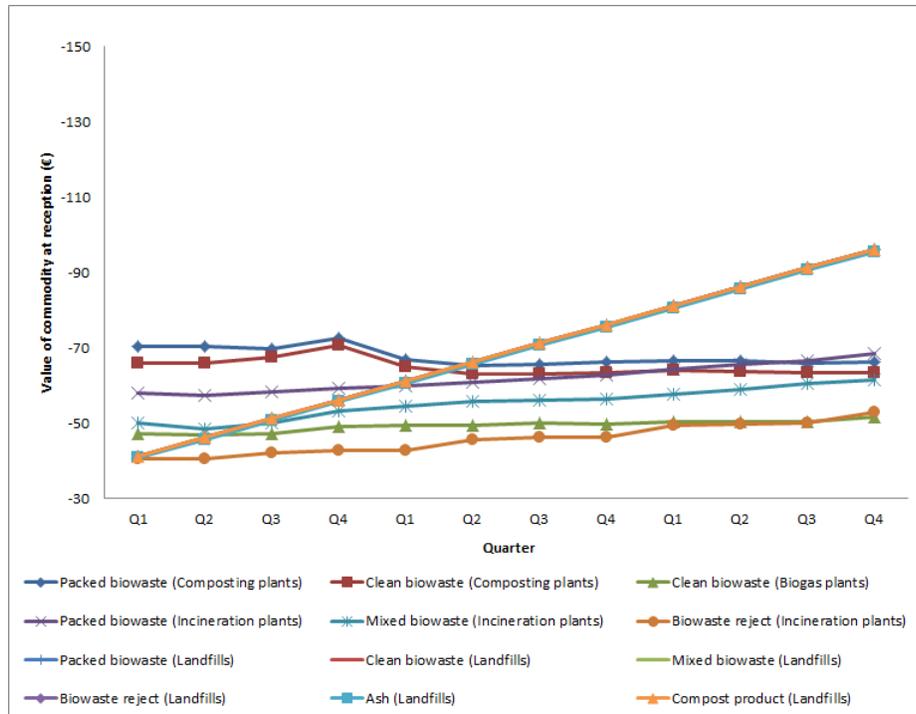
Figure 20: Average value of commodities at reception in euros per ton from different types of sites per quarter in scenario 2. All landfill gate fees overlap each other.
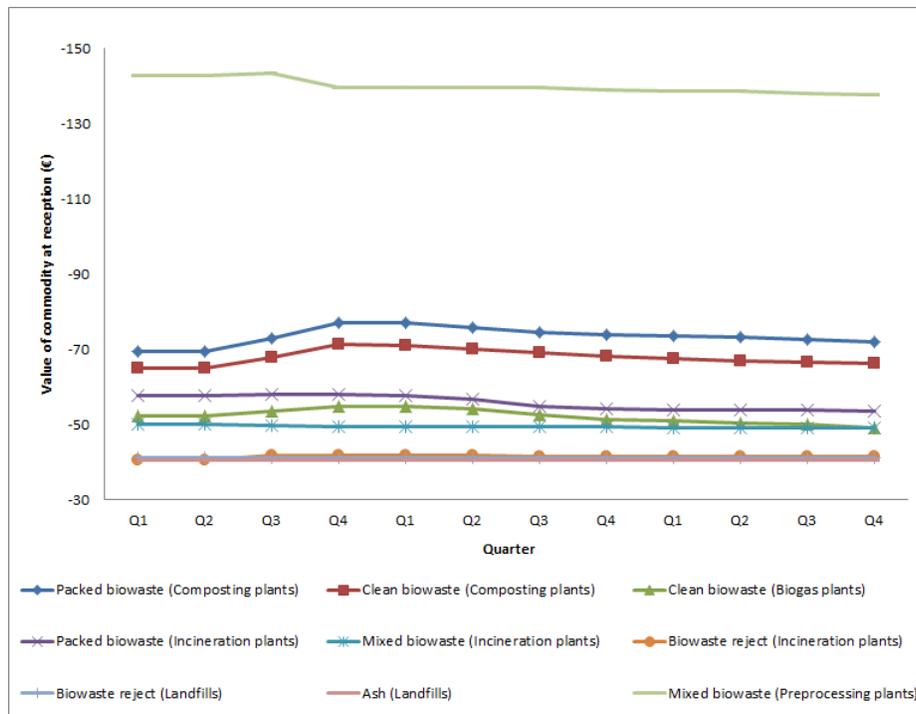
Figure 21: Average value of commodities at reception in euros per ton from different types of sites per quarter in scenario 3. All landfill gate fees overlap each other.
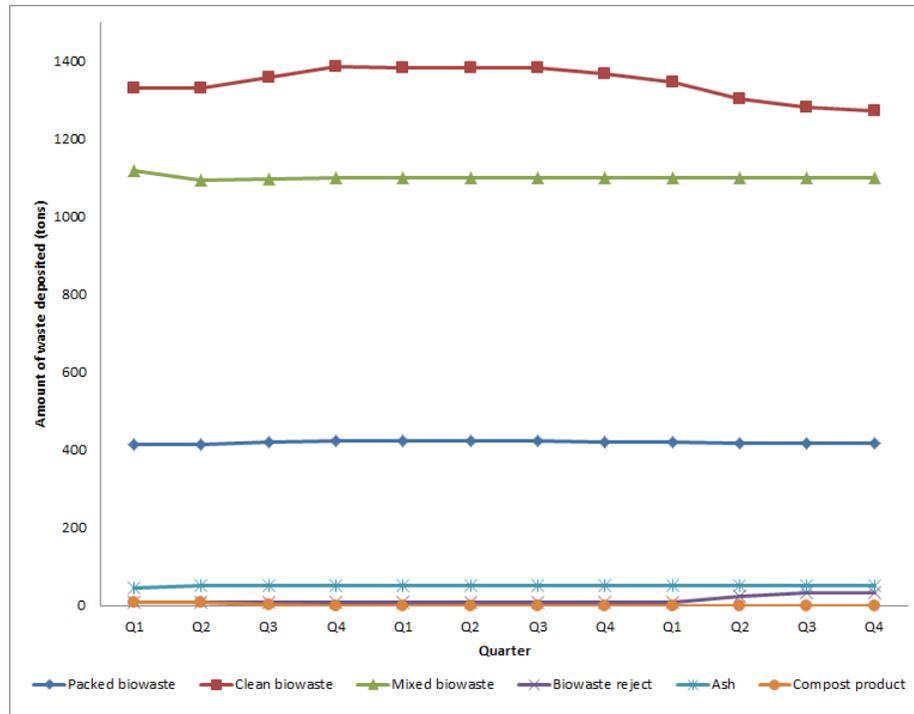
Figure 22: Commodities deposited to landfills per quarter in scenario 1.
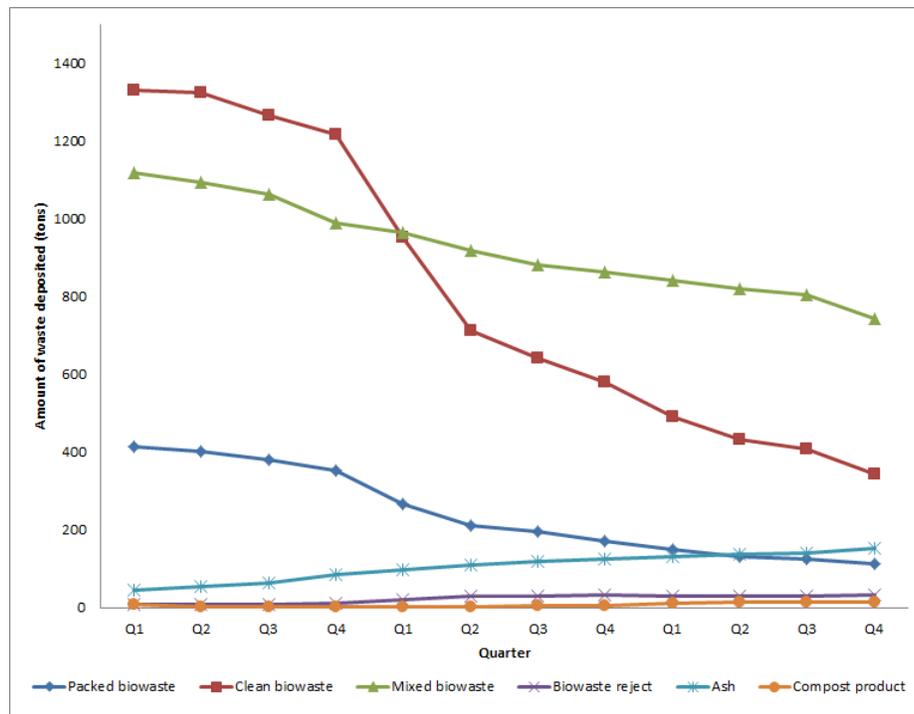
Figure 23: Commodities deposited to landfills per quarter in scenario 2.
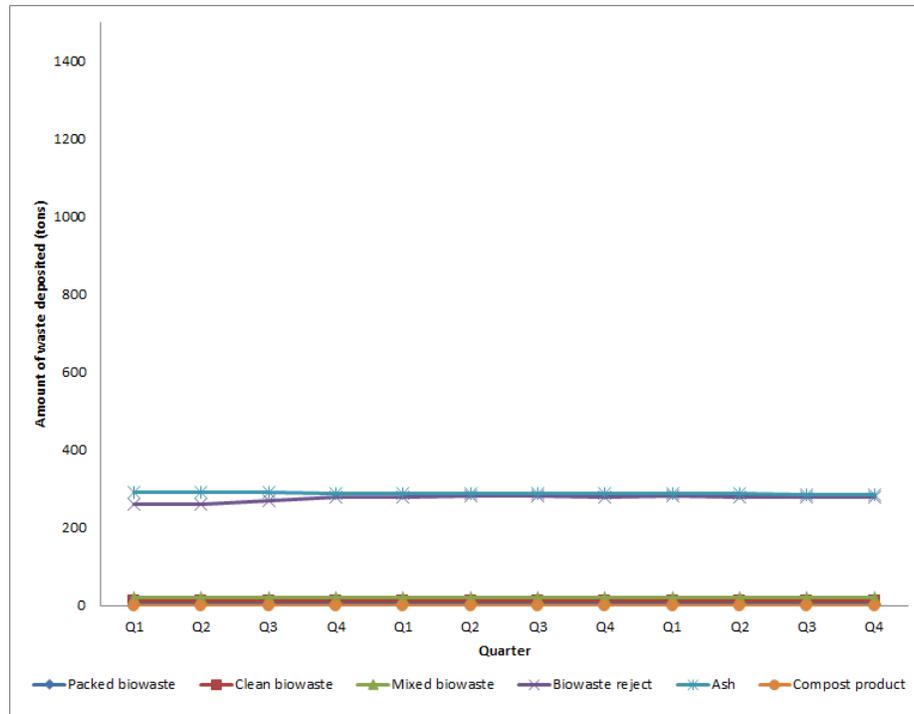
Figure 24: Commodities deposited to landfills per quarter in scenario 3.
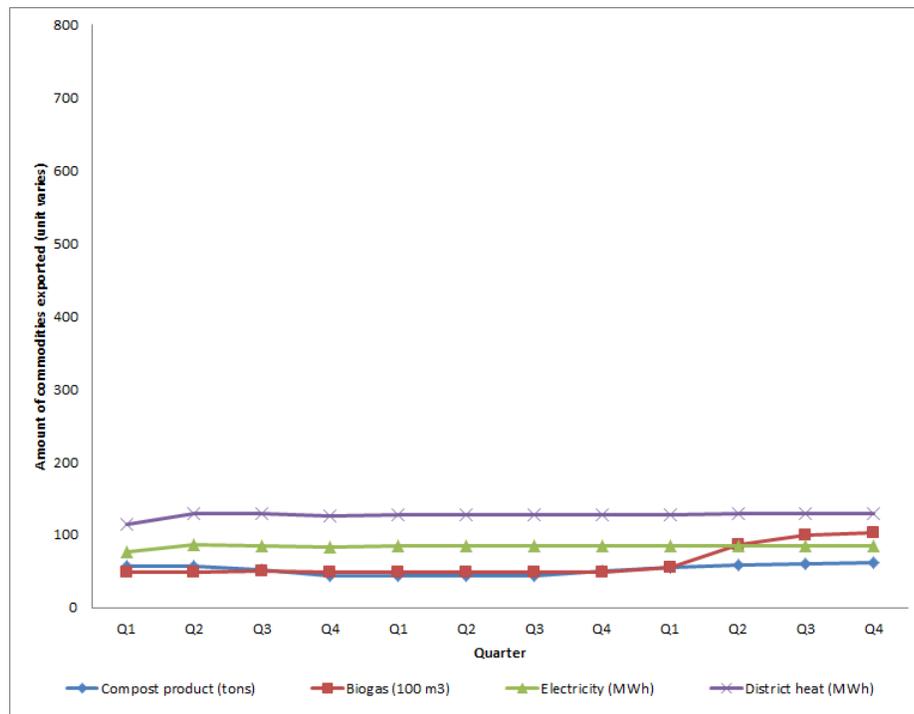
Figure 25: Commodities sold to markets per quarter in scenario 1.
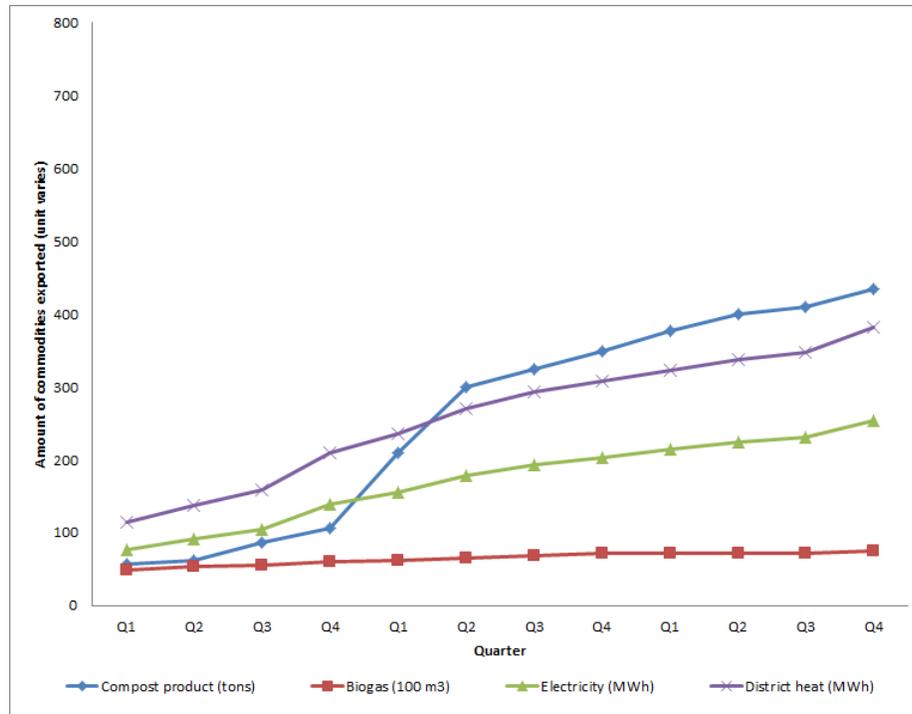
Figure 26: Commodities sold to markets per quarter in scenario 2.
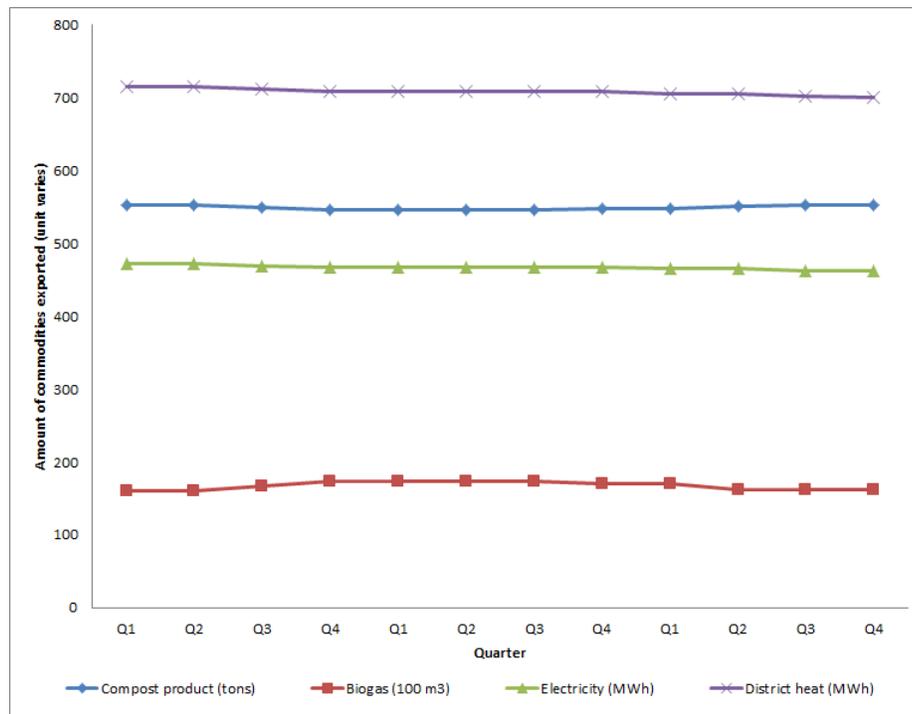
Figure 27: Commodities sold to markets per quarter in scenario 3.

6.8 Challenges in Validation

Proper validation of the model was not possible in the scope of this project, as there was not enough data available from the actual system. Credibly validating the model would have required both credible parameter data and credible trend data from the same time period. The parameter data is important for structural validity and the trend data for replicative validity.

The data that was used to parameterize the model was taken from many sources, as explained in section 5. The data from these sources does not always come from the target period or region of the model, which is the Finnish waste management industry from the year 2008. Thus, some of the parameter values are only rough approximations. This results in reduced structural validity of the model. To increase the structural validity, more credible data for the parameterization of the model would have to be collected.

Statistical data was readily available for only some of the trends that were of interest. Furthermore, the available statistical trends were not always coherent or comparable with each other, which made it difficult to make any conclusions based on them. In addition, the data was generally aggregated high level data. Detailed statistics would have been more useful for the validation. Because of these reasons, the replicative validity of the model could not be evaluated during this project. In order to do this, coherent and comparable statistical trend data should be collected, with smaller granularity if possible.

7 Feasibility Analysis

7.1 Overview

The feasibility of the ABMS paradigm in implementing WVCA models was evaluated based on the subjective experiences of the author during the modeling project and the anonymous feedback from experts. The feasibility is examined from the points of view of model design, implementation, verification, validation, credibility, experimentation, analysis of results, and comparison to alternative methods.

7.2 Feedback from Experts

After the second version of the implementation was sufficiently finished, experts from the areas of waste management, modeling and simulation were asked to evaluate the feasibility of the modeling method. This was done in order to get a wider perspective to the feasibility of the modeling method. The evaluation was done by giving the experts a presentation of the model, modeling method and the finished simulation program. The experts were allowed to freely ask questions during the presentation. After the presentation, the experts filled an anonymous feedback form. The feedback form and the received results were in Finnish. The translated feedback form and collected results can be found in appendix A.

The questionnaire contained 12 questions in total: 2 multiple-choice questions concerning the related experience of the answerer, 3 multiple-choice questions concerning the understandability of the presentation, one field for free comments regarding the presentation, two multiple-choice questions concerning the apparent feasibility of the modeling method, two questions concerning the apparent strengths and weaknesses of the modeling method, one question concerning alternative modeling methods, and one comment field concerning the model in general.

12 filled forms were collected from the participants, of which 3 contained some unfilled multiple-choice questions. These answers are omitted when the relevant results are presented.

The experts had overall a reasonable amount of experience either from one field or both. Only two participants had less than a year of experience from either field. The distribution of the experience of the participants is shown in figure 28.

The presentation was experienced as understandable. Three of the 12 participants stated that the presentation was 'Quite understandable', whereas nine stated it was 'Understandable'. The description of the modeling method and the system to be modeled were experienced as comprehensive. The distribution of the answers concerning comprehensiveness are shown in figure 29. The participants had commented that the presentation was good and clear and had enough time for questions and discussion.

The modeling method was experienced as quite feasible and the participants were quite certain about their opinion. Only two of the participants were uncertain or quite uncertain about their opinion. The distribution of the answers concerning the feasibility of the modeling method and the certainty of the participants' opinions is

shown in figure 30.

Figure 28: The experience of the participant from the fields of waste management and modeling and simulation. 1 = Not at all, 2 = A little (less than one year), 3 = Moderately (one to three years), 4 = Plenty (more than three years). Each dot represents one participant, overlapping dots have been slightly displaced. One participant had not answered to both questions.
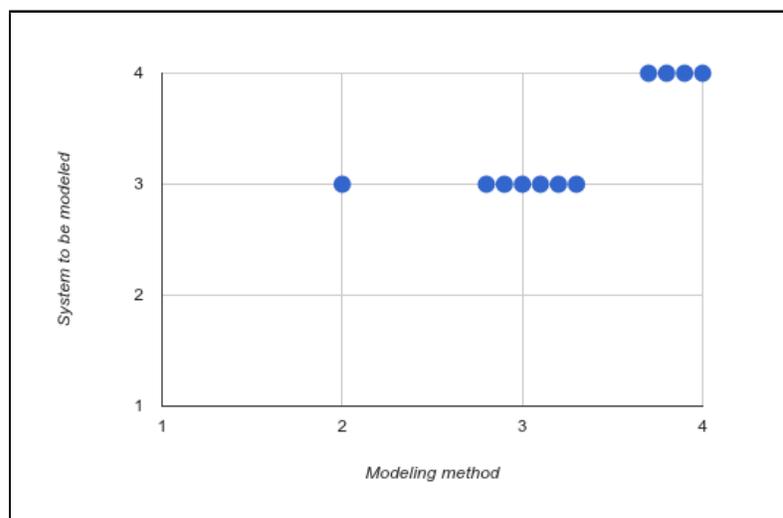
Figure 29: How comprehensive description did the presentation give of the modeling method and the system to be modeled. 1 = Limited, 2 = Quite limited, 3 = Quite comprehensive, 4 = Comprehensive. Each dot represents one participant, overlapping dots have been slightly displaced. One participant had not answered to both questions.
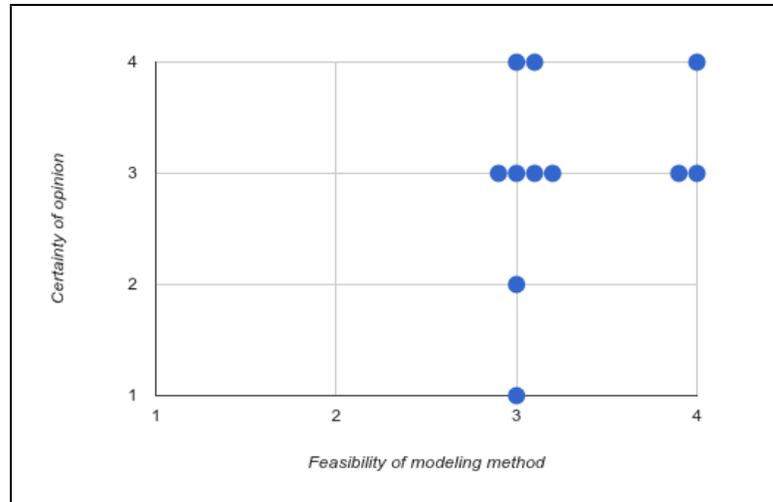
Figure 30: How feasible does the presented method appear to be and how certain the participants are of their opinion. 1 = Infeasible/Uncertain, 2 = Quite infeasible/Quite uncertain, 3 = Quite feasible/Quite certain, 4 = Feasible/Certain. Each dot represents one participant, overlapping dots have been slightly displaced. One participant had not answered to either question.

The participants experienced that the modeling method had certain strengths. They were:

- Parameterization and programmability of the model

- Ability to model decision-making in a situation where there are many competing actors

- Ability to give a good overall picture of the system

- Simplicity of the basic idea

- Scalability of the model

- Predictability of the amount of different data

- Ability to model complex systems

- Ability to model effects of single actors to entire system

- Making use of large amounts of background material

- Illustrativeness of the model

- Only viable way to model effects of political decisions

The participants also experienced that the method had certain weaknesses. They were:

- Defining the value of waste is difficult

- Timestep of the model compared to reality

- Part of initial data is difficult to collect

- Lack of reliable data makes validation and calibration difficult

- Not taking into account the differences in processing costs for different compositions of waste

- Understanding the behavior of the model

- The ability of other people to use the model

- Explaining the results of the model

The participants did not know of any other modeling method which could be better for modeling waste management systems from this perspective. However, some methods were said to be better for other types of modeling concerning waste management, such as processes and environmental effects.

The participants made some general comments concerning the modeling method. They were:

- It is not clear how long it takes to get to equilibrium

- It is not clear how long it takes to run the simulation

- Technical level of the model and required computational resources affect usability

- A good start that can be refined

- It should be considered how small units should be agents

- The model is probably better at giving answers to higher-level questions

- It appears to be possible to add almost anything to the model

- Generated and avoided emissions could be added to the model

- The flow of certain material could be modeled

- The model could be made into a product

7.3 Analysis

In general, ABMS seems to be a feasible way for implementing WVCA models. However, there are some considerations that should be taken into account. In this subsection the feasibility is examined from different points of view.

Feasibility in General

The experts considered the modeling method to be quite feasible and were quite certain of this opinion. They noted that the method has both notable strengths and drawbacks. However, they could not name any other method which could be better for this type of modeling.

The WVCA model could be implemented using the ABMS modeling paradigm. The simulation can produce results of relevant type, such as trend data concerning waste prices and transported ton-kilometers. The model can be parameterized using real world data, such as site locations, market prices and landfill gate fees. If this data would be available also from the real system, the model could be objectively validated at the system-level.

Design

The concepts used in ABMS correspond well with the concepts in WVCA. For example, the actors in WVCA correspond to the agents in ABMS, the trading of commodities and messages correspond with agent interactions, and the environmental effects of waste processing correspond with interactions between agents and environment. This makes it easy to transform the conceptual WVCA model into an implementable ABMS model.

As ABMS model design has only few limitations, various features can be included in the model. This allows the inclusion of, for instance, different types of regulations and behavior models. The model can be freely parameterized and the amount of agents is not limited. This allows the model to be applied to different types of waste management systems.

As the behaviors encountered in the real world can be complex, designing algorithms that are sufficiently similar can be difficult. Some behaviors can appear to be easy to define approximately, but prove to be difficult when an exact definition is required. For instance, the requirement that an agent should make economically rational decisions can be challenging to implement as an algorithm if the market situation is complicated.

The model may be better at answering higher level questions, although it also presents the possibility of examining the effects of single actors to the whole system. If needed, the model can be expanded to contain, for example, detailed models of processes or transportation, or the timestep can be made smaller. However, the amount of details also adds to the complexity of the model.

Implementation and Verification

Implementing and verifying ABMS models is fairly straightforward. If the model is defined precisely, the implementation should not bring along any special difficulties. The model can be implemented with the help of an ABMS toolbox, and verified using, for example, unit and integration testing along with VOMAS [35]. There are also many toolboxes available to help the implementation.

If the model is still experimental and does not have a precise definition, the implementation may be more challenging, as there may be changes to the model definition during implementation. However, this issue may be decreased if the model is implemented in phases of increasing complexity and detail.

Validation

The validation of ABMS models in general can be challenging, as explained in section 3. Furthermore, collecting relevant data for validation is currently challenging, as explained in section 6. Sources for this data could include, for example, national or regional statistics and statistics from individual companies. Not all of this data may be available for researchers, as there may not be sufficient public statistics and private companies may be unwilling to disclose accurate information about their actions. This may limit the availability of objective validation methods.

As the availability of objective validation methods may be limited, subjective validation will likely be required in addition. However, subjective validation is only as good as the available expert knowledge. As the model may aim to replicate many different behaviors, such as system-level trends and company-level decisions, experts from multiple different areas may be required to participate in the validation. This increases the cost and time required for the validation process.

Credibility

The credibility of these kinds of models is challenging to evaluate, as it depends not only on the model or the modeling approach, but also on the model end users. As the model was not yet used for any practical purpose, experiences from the end users could not be collected. These experiences would be important in evaluating the actual credibility.

In general, the model can be verified and validated, which is a good basis for credibility. If the data that is used to parameterize and validate the model is credible, and the validation procedure is documented, the credibility of the model increases. The functionality of the model can be fully documented, which may also increase the credibility.

As the model replicates complex behavior, it may be difficult for the user to understand why the model behaves in a certain way, which may decrease the credibility of the model if no explanation is found. Many simplifications have also been made in the model, which may reduce the credibility of the model if it is used for low-level simulation.

Experimentation and Analysis of Results

ABMS models can yield a large amount of diverse data, which is an excellent basis for thorough analysis. The model can also make use of a large amount of background data. As the amount of data is large, appropriate methods have to be used for storing the data and transforming it into a form that is suitable for further analysis.

The model appears to be able to produce reasonable behavior. However, as there may be a large amount of different events happening in the system, and the behavior of the system may be complex, it may be difficult to intuitively explain why the model behaves in a certain way. Still, as all the simulation events can be stored in a log file, the behavior can be analyzed afterwards if it raises any questions.

As there is a large amount of agents and events, the simulation may require large computational resources, especially if the model is very large, or if many simulation runs are required. This may limit the usability of the model, as it may take a long time to run these simulations using a personal computer. However, there are ways to distribute the computational load. In the case of a large model, it may be possible to distribute the model into smaller submodels, which could be run simultaneously on different platforms. However, this may make the model design more challenging. If a large amount of runs with the same model are needed, the runs could be done simultaneously on different platforms.

Comparison to Alternative Methods

The author is not aware of any alternative methods for doing this type of modeling. On the other hand, the experts could not either identify any method that would be better than ABMS. Therefore, it is possible that this is currently the only method that is able to model the distributed decision making in a waste management system.

8 Conclusions

In this section a summary of the thesis is presented, followed by recommendations for the further development of the model.

8.1 Summary

Modeling waste management systems is important for the following of environmental effects and the development of regulations. Many types of models exist for this purpose, but they have known limitations. Waste value chain analysis is a new method for modeling waste management systems. In this thesis, the feasibility of using agent-based modeling and simulation for implementing waste value chain analysis models was studied.

Simulation modeling is a common way to create executable models of different types of systems. Simulation modeling project is composed of different phases, which are the planning, definition, implementation, validation, experimentation, analysis and presentation phases. The project workflow is not necessarily linear, but may return to an earlier phase when necessary. Verification, validation and credibility are important concepts in simulation modeling.

Agent-based modeling and simulation is a method for modeling systems with distributed decision-making. The decision-makers are modeled using agents, which interact with each other and the environment. The simulation results are aggregated from the individual events that have happened during the simulation. The advantage of this kind of constructive model is, that the system-level behavior needs to be defined only implicitly, which may make the modeling of complex behavior possible without having to define explicit functions for the behavior. The disadvantage is, that it brings additional challenges for the validation of the model.

In the waste value chain model, the different actors in the system are modeled individually. The actors can be divided into generating, processing and depositing actors. The processing actors have processes, which they use to transform one type of commodity into others. Waste and other commodities are transported in the system based on agreements that the actors make with each other. The behavior of the actor is decided by the actor's behavior model.

An experimental model of a real system was created in order to study the feasibility of the modeling paradigm for implementing these kinds of models. The system being modeled is the waste management of Finnish daily consumables retail shops. To limit the scope of the model, only biowaste related commodities were considered. Other simplifications were also made to make the modeling effort easier. Data for the model was collected from many sources.

The model was implemented using the Repast Simphony 2.0 beta toolkit. The implemented model was able to produce results that could have been validated, if sufficient data would have been available. Validation of the model would have required a large amount of different kinds of data from the system. However, this data could not be collected within the scope of this project.

To get a broader view of the model feasibility, the model was presented to a

group of experts and anonymous feedback was collected. The experts found the model to be quite feasible and were quite certain in this opinion.

The feasibility of the modeling method was analyzed from different perspectives. Overall, the method seems to be feasible. The method is very flexible and allows the inclusion of many different types of features. The simulation results can create a lot of data, which is a good basis for analysis. However, the validation of these kinds of models may be challenging, as it can require a considerable amount of data to be collected.

8.2   Future Development

The model is promising as a tool for development of regulations for waste management systems. However, the model is still incomplete, and further development is required before the model can be used.

Some of the features of the simulation model could have been implemented differently, depending on how the conceptual model is interpreted. To enable the model to be replicated and analyzed, the WVCA model definition should be made more exact.

As the model definition was subject to change during the implementation phase, no formal verification procedure was used, but the verification was done manually. To reduce the time needed for manual verification in the future, a structured way to verify the model implementation should be considered.

The validation of the model requires that the exact behaviors of interest to be defined, and relevant data to be collected. If there is not enough data on some aspects, subjective validation methods have to be used for validating these behaviors.

The actual users of the model should be defined and taken into account when developing the model further. This could help to increase the credibility of the model and direct the development to better meet the end users' needs.

References

[1] A. Law and D. Kelton, *Simulation Modeling and Analysis.* McGraw-Hill, 1991.

[2] Tilastokeskus, "Maakunnat 2012." `http://www.tilastokeskus.fi/meta/luokitukset/maakunta/001-2012/index.html` (Accessed 12.10.2012), 2012.

[3] T. H. Christensen, ed., *Solid Waste Technology & Management.* Blackwell Publishing, 2010.

[4] "Directive 2008/98/EC of the European Parliament and the Council of 19 November 2008 on waste and repealing certain Directives (Text with EEA relevance)." OJ L 312, pp. 3-30, 2008.

[5] Ympäristöministeriö, "Jätelaki 646/2011," 2011.

[6] A. Morrissey and J. Browne, "Waste management models and their application to sustainable waste management," *Waste Management*, vol. 24, no. 3, pp. 297–308, 2004.

[7] N. Hanley and C. L. Spash, *Cost-Benefit Analysis and the Environment.* Edward Elgar, 1993.

[8] W. Klöpffer, "Life cycle assessment," *Environmental Science and Pollution Research*, vol. 4, no. 4, pp. 223–228, 1997.

[9] B. Roy and D. Vanderpooten, "The european school of MCDA: Emergence, basic features and current works," *Journal of Multi-Criteria Decision Analysis*, vol. 5, no. 1, pp. 22–38, 1996.

[10] C. Macal and M. North, "Tutorial on agent-based modeling and simulation," *Journal of Simulation*, 2010.

[11] G. Conzelmann, G. Boyd, V. Koritarov, and T. Veselka, "Multi-agent power market simulation using EMCAS," in *Power Engineering Society General Meeting, 2005. IEEE*, pp. 2829 – 2834 Vol. 3, 2005.

[12] R. Y. Rubinstein and D. P. Kroese, *Simulation and the Monte Carlo Method.* John Wiley & Sons, 2008.

[13] G. S. Fishman, *Discrete-Event Simulation.* Springer, 2001.

[14] R. J. Brooks and A. M. Tobias, "Choosing the best model: Level of detail, complexity, and model performance," *Mathematical Computer Modelling*, vol. 24, no. 4, pp. 1–14, 1996.

[15] R. G. Sargent, "Verification and validation of simulation models," in *Proceedings of the 37th Conference on Winter Simulation*, pp. 130–143, 2005.

[16] B. P. Zeigler, H. Praehofer, and T. G. Kim, *Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems.* Academic Press, second ed., 2000.

[17] M. Pidd, "Five simple principles of modelling," in *Proceedings of the 1996 Winter Simulation Conference*, pp. 721–728, 1996.

[18] C. Macal and M. North, "Agent-based modeling and simulation: Desktop ABMS," in *Proceedings of the 2007 Winter Simulation Conference*, pp. 95–106, 2007.

[19] J. H. Miller and S. E. Page, *Complex Adaptive Systems: An Introduction to Computational Models of Social Life.* Princeton University Press, 2007.

[20] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," *Computer Graphics*, vol. 21, no. 4, pp. 25–34, 1987.

[21] J. M. Epstein and R. Axtell, *Growing Artificial Societies: Social Science From the Bottom Up.* Brookings Institution Press, 1996.

[22] C. Deissenberg, S. van der Hoog, and H. Dawid, "EURACE: A massively parallel agent-based model of the european economy," *Applied Mathematics and Computation*, vol. 204, no. 2, pp. 541–552, 2008.

[23] J. L. Segovia-Juarez, S. Ganguli, and D. Kirschner, "Identifying control mechanisms of granuloma formation during M. tuberculosis infection using agent-based model," *Journal of Theoretical Biology*, vol. 231, no. 3, pp. 357–376, 2004.

[24] K. M. Carley, D. B. Fridsma, E. Casman, A. Yahja, N. Altman, L. Chen, B. Kaminsky, and D. Nave, "BioWar: Scalable agent-based model of bioattacks," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 36, no. 2, pp. 252–265, 2006.

[25] H. S. Nwana and D. T. Ndumu, *Agent Technology: Foundations, Applications and Markets*, ch. 2: A brief introduction to software agent technology. Springer, 2002.

[26] K. Z. Haigh and M. M. Veloso, "Planning, execution and learning in a robotic agent," in *AIPS 1998 Proceedings*, pp. 120–127, 1998.

[27] J. Holland, *Hidden Order: How Adaptation Builds Complexity.* Perseus books, 1995.

[28] L. Testafatsion and K. Judd, *Handbook of Computational Economics.* Elsevier, 2006.

[29] L. Walras, *Elements of Pure Economics.* Irwin, 1954.

[30] L. Chwif, M. R. P. Barretto, and R. J. Paul, "On simulation model complexity," in *Proceedings of the 2000 Winter Simulation Conference*, pp. 449–455, 2000.

[31] M. J. North, N. T. Collier, and J. R. Vos, "Experiences creating three implementations of the Repast agent modeling toolkit," *ACM Transactions on Modeling and Computer Simulation*, vol. 16, no. 1, pp. 1–25, 2006.

[32] S. Luke, C. Cioffi-Revilla, L. Panait, K. Sullivan, and G. Balan, "MASON: A multiagent simulation environment," *Simulation*, vol. 81, no. 7, pp. 517–527, 2005.

[33] S. Tisue and U. Wilensky, "NetLogo: A simple environment for modeling complexity," in *International Conference on Complex Systems*, 2004.

[34] F. Klügl, "A validation methodology for agent-based simulations," in *Proceedings of the 2008 ACM symposium of Applied computing*, pp. 39–43, 2008.

[35] M. A. Niazi, A. Hussain, and M. Kolberg, "Verification & validation of agent based simulations using the VOMAS (virtual overlay multi-agent system) approach," in *Proceedings of the Second Multi-Agent Logics, Languages, and Organisations Federated Workshops*, 2009.

[36] G. Tchobanoglous, H. Theisein, and S. A. Vigil, *Integrated Solid Waste Management: Engineering Principles and Management Issues*. McGraw-Hill, 1993.

[37] R. Veranen, "Lehdistötiedote päivittäistavarakaupan myymälärekisteri 2008." `http://www.fi.nielsen.com/site/documents/Lehdistotiedote_18March2009.pdf` (Accessed 02.07.2012), 2009. Press release.

[38] K. Koistinen and T. Vesala, "Päivittäistavarakaupan rakennemuutos suomen keskeisillä kaupunkiseuduilla 1995 - 2003." `http://www.kuluttajatutkimuskeskus.fi/files/4931/2006_06_julkaisu_paivittaistavara.pdf` (Accessed 02.07.2012), 2006. National Consumer Research Centre, publications 6:2006.

[39] Jätelaitosyhdistys, "Tietoa kuntien jätehuollosta 2010: Kiinteistökohtaisen keräyksen jätemaksutiedot ja jätelajien vastaanottohinnat." `http://jly.fi/jatemaksut2010.pdf` (Accessed 12.10.2012), 2011.

[40] T. Myllymaa, K. Moliis, A. Tohka, P. Rantanen, M. Ollikainen, and H. Dahlbo, "Jätteiden kierrätyksen ja polton käsittelyketjujen ympäristökuormitus ja kustannukset." `http://www.ymparisto.fi/default.asp?contentid=298205&lan=fi` (Accessed 12.10.2012), 2008.

[41] Päivittäistavarakauppa ry, "Päivittäistavarakauppa ja ympäristö, katsaus päivittäistavarakaupan ympäristövaikutuksiin vuonna 2003." `http://www.pty.fi/fileadmin/pty_tiedostot/Julkaisut/Ymparistoraportti.pdf` (Accessed 04.07.2012), 2003.

[42] Helsingin seudun ympäristöpalvelut, "Petra-jätevertailu." `http://www.petrajatevertailu.net/hsy/?mo=ostats&view=4&type=1&tol=1&gid=30&y=2008` (Accessed 12.10.2012), 2012.

[43] J. Laaksonen, A. Kangasrääsiö, and J. Kaila, "Feasibility of agent based simulation for modeling the decision making processes in recycling and its effects on material flows and environmental impacts," in *International Conference on EcoBalance 2012*, 2012.

[44] J. Swaminathan, S. Smith, and N. Sadeh, "Modeling supply chain dynamics: A multiagent approach," *Decision Sciences*, 1998.

[45] D. J. van der Zee and J. G. A. J. van der Vorst, "A modeling framework for supply chain simulation: Opprtunities for improved decision making," *Decision Sciences*, 2005.

[46] Argonne National Laboratory, "Repast." `http://repast.sourceforge.net`.

[47] George Mason University, "MASON." `http://cs.gmu.edu/~eclab/projects/mason/`.

[48] U. Wilensky, "NetLogo." `http://ccl.northwestern.edu/netlogo/`.

[49] Swarm Development Group, "Swarm." `http://www.swarm.org`.

[50] C. Nikolai and G. Madey, "Tools of the trade: A survey of various agent based modeling platforms," *Journal of Artificial Societies and Social Simulation*, vol. 12, no. 2, 2009.

[51] The Eclipse foundation, "Eclipse." `http://www.eclipse.org`.

A Feedback Form and Collected Results

The feedback form, translated from Finnish:

A questionnaire concerning the feasibility of agent based modeling for modeling waste value chains

The purpose of this questionnaire is to measure the feasibility of agent based modeling for modeling waste value chains by collecting expert opinions. The results of this questionnaire are used in the further development of the model and in my master's thesis to support the analysis. The questionnaire is answered anonymously.

Circle the most appropriate option or answer in writing.

Questions concerning the answerer

1. How much experience do you have from the field of waste management?

   1 Not at all
   2 A little (less than one year)
   3 Moderately (one to three years)
   4 Plenty (more than three years)

2. How much experience do you have from the field of modeling and simulation?

   1 Not at all
   2 A little (less than one year)
   3 Moderately (one to three years)
   4 Plenty (more than three years)

Questions concerning the presentation

3. How understandable the presentation was in your opinion?

   1 Difficult to understand
   2 Quite difficult to understand
   3 Quite easy to understand
   4 Easy to understand

4. How comprehensive was the description of the modeling method in your opinion?

   1 Limited
   2 Quite limited
   3 Quite comprehensive
   4 Comprehensive

5. How comprehensive was the description of the system to be modeled in your opinion?

   1 Limited
   2 Quite limited
   3 Quite comprehensive
   4 Comprehensive

6. Other comments concerning the presentation

   _____

   _____

Questions concerning the model

7. How feasible is the presented modeling method for modeling waste value chains in your opinion?

   1 Infeasible
   2 Quite infeasible
   3 Quite feasible
   4 Feasible

8. How certain are you of your opinion?

   1 Uncertain
   2 Quite uncertain
   3 Quite certain
   4 Certain

9. What are the strengths of the presented modeling method in your opinion?

   _____

   _____

10. What are the weaknesses of the presented modeling method in your opinion?

    _____

    _____

11. Do you know any other modeling method that could be better for modeling waste value chains? What could be the advantages of this method compared to the presented method?

    _____

    _____

12. Other comments concerning the model

    _____

    _____

    Thank you for your answers

N	1	2	3	4	5	6	7	8	9	10	11	12
1	3	2	4	3	3	-	3	3	Parameterization, programmability	Defining the value of waste. Timestep of simulation in relation to reality, can changes happen faster in reality?	No	Length of simulation, how long does it take to get to equilibrium? How long does it take to run the simulation? Technical level of the model and needed computational resources affect the usability.
2	4	1	4	3	3	-	3	3	Quite good overall picture	Part of initial data is difficult to collect	-	-
3	4	4	4	4	4	Clear presentation	4	4	Modeling decision making in a system, which has many competing decision makers	Lack of reliable data makes validation and calibration difficult	For modeling decision making, no. For modeling other properties (processes, material flows, environmental effects, etc.), yes.	A very promising model

Table 2: Collected results from the questionnaire, translated from Finnish, part 1

N	1	2	3	4	5	6	7	8	9	10	11	12
4	4	2	4	3	3	-	3	3	Gives a good picture of the overall system and the effects of changes	Taking into account the differences in processing costs for different compositions of waste, the model uses average prices. Building this kind of model may be challenging in practice.	-	Overall a good start that can be refined.
5	4	4	4	3	3	Good presentation.	3	4	The basic idea is simple. The number of agents is not limited.	Getting the price formation of processing closer to reality is difficult. However, this is not only the problem of this method.	None known.	It should be considered how small units should be agents. A single shop is quite small. The model is probably better at giving answers to a bit higher level questions.

Table 3: Collected results from the questionnaire, translated from Finnish, part 2

N	1	2	3	4	5	6	7	8	9	10	11	12
6	3	2	3	2	3	There was enough time for questions and discussion.	3	3	The predictability of the amount of different data.	The lack of validation data.	Not for decision making; MFA STAN (TO VI-ENNA) - could they be merged?	Good luck!
7	1	2	4	3	3	-	3	1	-	-	-	-
8	2	4	3	4	4	-	-	-	-	-	-	-
9	2	4	4	3	-	-	3	2	The modeling of complexity.	Understanding the behavior of the model, validation.	-	-
10	1	2	4	4	4	The presentation made the subject clearer.	4	3	A good way to present abstract structures and the effects of single actors to the whole system.	Modeling a single process accurately. Difficult to be compared with the real world.	No.	-

Table 4: Collected results from the questionnaire, translated from Finnish, part 3

N	1	2	3	4	5	6	7	8	9	10	11	12
11	4	-	3	3	3	-	3	4	-	-	-	-
12	4	1	4	4	4	Good and clear presentation	4	3	Making use of large amounts of background material. Illustrativeness. No other real ways to model the effects of political decisions as clearly.	Repeatability, meaning the ability of other people to use the model. But is there a need for this is another question. Explaining the results of the model, it might not always be easy to understand what causes certain behaviors.	-	It seems to be possible to add anything to the model, one option would be to add the emissions and avoided emissions (e.g. behavior of energy plants), etc. Or model the flow of some metal or similar commodity. The model could also be made as a product, demand might exist.

Table 5: Collected results from the questionnaire, translated from Finnish, part 4