BACHELOR'S THESIS
Nima Ahmady-Moghaddam

Comparing Theories of Human Behaviour by Implementing them in MARS Agents: An Interdisciplinary Approach Based on the HuB-CC Framework

FACULTY OF COMPUTER SCIENCE AND ENGINEERING
Department Computer Science
Fakultät Technik und Informatik
Department Informatik

HAMBURG UNIVERSITY OF APPLIED SCIENCES
Hochschule für Angewandte Wissenschaften Hamburg

Nima Ahmady-Moghaddam

Comparing Theories of Human Behaviour by Implementing them in MARS Agents: An Interdisciplinary Approach Based on the HuB-CC Framework

Bachelor's Thesis submitted as part of the Bachelor's Exam
in the degree programme *Bachelor of Science Applied Computer Science*
at the Department Computer Science
of the Faculty of Computer Science and Engineering
of the University of Applied Sciences Hamburg

First supervisor: Thomas Clemen, PhD
Second supervisor: Maja Schlüter, PhD

Submitted on: 14 January 2022

Nima Ahmady-Moghaddam

Title of Thesis

Comparing Theories of Human Behaviour by Implementing them in MARS Agents: An Interdisciplinary Approach Based on the HuB-CC Framework

Keywords

Distributed artificial intelligence; Agent-based modelling, Theory-based behaviour modelling; Cognitive science; Socio-ecological system; Interdisciplinarity

Abstract

This thesis is concerned with the implementation of theories of human behavior as well as an agent architecture informed by domain expertise in an agent-based model written in MARS. A conceptual modeling framework called HuB-CC was used to identify, classify, and select the theories. The modeling approach was developed in an interdisciplinary exchange with the authors of the HuB-CC framework. The model results as well as the quality of the designed architecture are analyzed and potential avenues of further inquiry are explored.

Nima Ahmady-Moghaddam

Thema der Arbeit

Vergleiche von Theorien menschlichen Verhaltens durch ihre Implementierung in MARS Agenten: Ein interdisziplinärer Ansatz, basierend auf dem HuB-CC Rahmenwerk

Stichworte

Verteilte künstliche Intelligenz; Agentenbasierte Modellierung; Theoriebasierte Verhaltensmodellierung; Kognitionswissenschaft; Sozioökologisches System; Interdisziplinarität

Kurzzusammenfassung

Diese Arbeit befasst sich mit der Implementierung von Theorien menschlichen Verhaltens sowie einer durch Domänenexpertise informierten Architektur in einem agentenbasierten Modell, welches in MARS geschrieben ist. Zur Identifizierung, Einordnung und Auswahl der Theorien wurde ein konzeptionelles Modellierungsrahmenwerk namens HuB-CC verwendet. Der Modellansatz ist in einem interdisziplinären Austausch mit den Autorinnen des HuB-CC Frameworks entstanden. Die Modellergebnisse sowie die Güte der entworfenen Architektur werden analysiert und mögliche Vertiefungsrichtungen dieser Arbeit werden erörtert.

Acknowledgements

I would like to thank my first supervisor, Professor Thomas Clemen, for giving me the initial impulse that I needed to formulate the research question of this work and for consistently offering guidance throughout the development and writing process.

Likewise, I wish to thank my second supervisor, Professor Maja Schlüter, for helping me appreciate the interdisciplinary aspect of this work, facilitating an invaluable connection between me and the other co-authors of the HuB-CC framework, and offering insights and assistance on numerous occasions.

I would also like to thank Professor Sara Constantino and Professor Elke Weber for your support while I was sharpening and specifying the research question. The insights I gained through our conversations helped me to better navigate the domain-specific portions of this work.

I also wish to thank everyone at the MARS Group who contributes to the continued development of the modelling framework used for this work. Your efforts make works like the present one possible.

Lastly, I thank my parents. You have loved and supported me unconditionally and wholeheartedly since before I can remember and continue to do so beyond what I feel I deserve. I am grateful for having you in my life.

The real question is not whether machines think but whether men do.

The mystery which surrounds a thinking machine already surrounds a thinking man.

—B.F. Skinner (1969)

Contingencies of Reinforcement: A Theoretical Analysis

Contents

List of Figures

List of Tables

Abbreviations

ABM Agent-Based Models.

C# C Sharp.

CPR Common-Pool Resource.

CSV Comma-Separated Values.

DAI Distributed Artificial Intelligence.

GIS Geographic Information System.

HAW Hamburg University of Applied Science Hamburg.

HuB-CC Human Behavior-Cognition in Context.

IDE Integrated Development Environment.

JSON JavaScript Object Notation.

MARS Multi-Agent Research and Simulation.

MAS Multi-Agent Systems.

MDP Markov Decision Process.

MoHub Modelling Human Behaviour.

MRP Markov Reward Process.

OOP Object-Oriented Programming.

SES Social-Ecological Systems.

UML Unified Modelling Language.

1 Introduction

Human decision-making and behaviour is complex and understanding it better is of significant interest, as indicated by various scientific conferences and research studies that are dedicated to its analysis in different contexts (e.g., [13, 32, 42, 28]). Since humans have evolved to produce massive communities and social networks that span the Earth, it is apparent that the impact of their behaviour on themselves and on their environment is far-reaching [20, 3]. While studying human behaviour in the field (i.e., with real-world research subjects) might be the greatest potential source of insight into its nature and patterns, such studies can be time-intensive, expensive, difficult to coordinate, and prone to subjective influences from both the researchers and the subjects [22, 39]. There appears to be a need for and a benefit in exploring alternative, yet similarly promising ways of inquiry.

Over the last several decades, the study of human behaviour has experienced an increase in the use of computer-driven and automated models as analytical tools. This trend can be observed across many scientific disciplines and research areas, and is fuelled by an increasing abundance of technical innovation and computational power. While there are several types of modelling approaches, the agent-based modelling paradigm has received wide-spread attention for its ability to describe the individual and collective behaviour of a group of agent instances, each acting autonomously within a given environment [1, 17, 21, 30]. Such models have been developed in the context of food security [14], infectious disease [11], traffic planning [27], and disaster management [12], among many others. All of these examples, by virtue of their respective domain, are focused on modelling the behaviour of humans. However, in many cases, the assumptions and premises underlying these models tend to be too static and low-faceted to adequately describe the dynamic and varied behaviours of humans [33]. In models of Social-Ecological Systems (SES), particularly, the behavioural logic of human agents is often based on the theory of Rational Choice. According to this theory, a human actor's sole motivation when weighing options and making decisions is the optimisation of personal utility, i.e., the maximisation of some quantity that is of personal interest to her [40, p. 8–9]. While such a single-minded motivation might apply in some cases, empirical evidence strongly suggests that decision-making processes are far more nuanced in their considerations [32, 24]. One idea for improving ABM that involve human actors is to integrate insights from theories of human behaviour into them [34]. The intention is to enable agents to more closely represent their real-world counterparts.

This thesis describes the integration of three theories of human behaviour into an ABM. The main focus is on the selection of a well-suited model scenario, the identification of applicable theories of human behaviour, and the design as well as the implementation of the agents' structure and behavioural routine. The research question is how well the theories can be integrated into the model – given the available tools and means – and how they manifest themselves in the agents' behaviour upon executing the model. To facilitate this effort conceptually, the HuB-CC framework is used. This is a framework for human behaviour modelling that aims to inform design choices of modellers by offering a theoretical foundation. Furthermore, it provides a way to systematise the modelling process by stratifying the human decision-making process into distinct cognitive components and suggesting relationships between them. An overview of these concepts is provided in this thesis. For the technical realisation, the MARS framework is used. This is an agent-based modelling framework that uses Object-Oriented Programming (OOP) to write models. The descriptiveness and some of the structural capabilities of the OOP paradigm will be explored with respect to the agents' internal structure.

Thesis Outline The outline for the remainder of this thesis is as follows. Chapter 2 explores related works and concepts that are deemed relevant for the subsequent chapters. Chapter 3 outlines the domain-specific and technical requirements of the intended model. Chapter 4 describes the MARS framework, the HuB-CC framework, the conceptual basis of the model, and the theories chosen to model the agents' behaviour. Chapter 5 describes in detail the design and implementation process of the model as well as the formalisation of the theories and their integration into the model. Chapter 6 illustrates the functionality of the model by presenting model outputs and results. Lastly, Chapter 7 offers a critical discussion of the results, an analysis of the implemented agent structure, a general review of the chosen approach (particularly with respect to its interdisciplinary nature), and an outlook to potential next steps that might follow this work.

2 Concepts and Related Work

As mentioned in Chapter 1, human behaviour modelling has become increasingly popular in the behavioural and social sciences. The agent-based modelling paradigm has received particularly high attention. However, there are also other modelling approaches that should at least be mentioned briefly in this work. This section gives a general overview of some of these approaches and outlines the conceptual framework for the remainder of the thesis.

The exercise of modelling human behaviour encompasses the description of characteristics of that behaviour in calculable and executable models. By using such models, researchers wish to study and analyse properties of human behaviour to develop a better understanding of it. Insights gained during the modelling process and from model outputs can, ideally, be transferred to recommendations for actionable change in the real world. Models of SES, for example, study the impact of human behaviour on the natural environment. A particularly acute application of models of SES is the identification of individual and/or communal behaviours that might facilitate or hinder the achievement of sustainability and climate change goals.

There are different types of models from different disciplines that can serve such a purpose. Systems theory [23], for example, tries to model human behaviour as a set of differential equations, where solutions of the system fulfil all equations simultaneously and represent some desirable system state. In control theory [38], a Markov Reward Process (MRP) can be used to model a chain of states that are connected via transitions, each of which is associated with a reward. The solution to such a system involves a path through the states that maximises a reward function. A MRP can be extended to become a Markov Decision Process (MDP) by adding a set of possible actions to the system. An agent that navigates such a system has these actions at its disposal and must choose them in such an order so that, again, a given reward function is maximised. A performed action is indicative of decision made by the agent: if an agent in a given state has two options and chooses one over the other, that choice reveals a decision and a preference regarding the state-action pairs that are available to the agent in its current state. MDPs can therefore be used to model decision processes and preference patterns.

The use of game theory [40] has been especially popular in economic models to study the rationale for certain decisions in economic transactions. A game-theoretic model is a model in which rational agents are required to maximise some function while engaging in strategic interactions with other rational agents. To converge to a solution, agents need to find a system equilibrium. Due to the involvement of strategic interactions, game theory lends itself to modelling cooperative and competitive settings as well as other settings that required some form of negotiation between different actors. While game-theoretic models have seen widespread application, the use of rational agents has been criticised for not adequately capturing the possible considerations that human decision-makers might make in different contexts. Rationality is defined here as acting in one's own self-interest. Therefore, a rational actor, as mentioned in Chapter 1, is one that acts in its own self-interest. In game-theoretic settings, this translates to maximising personal utility – typically at the cost of all other factors.

Since its conception, the theory of Rational Choice has been augmented in a few ways. One prominent extension was made by Herbert A. Simon, who defined the theory of Bounded Rationality[1] [36]. Like the theory of Rational Choice, this theory is concerned with describing evaluation and decision processes of a rational actor. But it takes into account that such an actor is in some way limited. The limitations are imposed either by the intrinsic nature of the agent, the environment the agent is situated in, or both. With the advent of Multi-Agent Systems (MAS) and ABMs, an opportunity presented itself to model more realistic human agents. One reason for this is that these systems and models are part of the overarching concept of Distributed Artificial Intelligence (DAI): a family of intelligent systems in which intelligence is not centralised (i.e., located at one and only one location), but rather virtually or physically distributed across multiple entities. The decentralisation of intelligence enables the possibility of implementing bounded agents (i.e., ones with incomplete knowledge and/or capabilities) and possibly ones that do not act rationally under all circumstances.

As indicated in Chapter 1, ABMs are used in a wide range of fields and disciplines to study different behavioural phenomena. This thesis focuses on the application of ABMs to model SES. In this domain, ABMs have recently been used to model irrigation games [25], pastoral systems [16], and fisheries [42], among others. At the conceptual level, ABMs

[1] A more detailed elaboration of this theory is given later in the thesis.

have been used to, for example, compare different formalisations of the same conceptual behaviour description (rational and altruistic) [29]. The notion of incorporating aspects from theories of human behaviour is common to all of these research efforts. To this end, the theories need to be formalised [34]. This means that the usually informal descriptions in which the essence of the theory is conveyed need to be transformed into a form that lends itself to being used for creating a design and an implementation plan for software agents.

3 Requirements

This Chapter features a list of functional and non-functional requirements of the model that is implemented for this thesis. The requirements are derived from the outline of the goal presented in Chapter 1 and the main concepts and related works presented in Chapter 2.

1. **Identify a model basis:** Since one of the goals of this work is to implement a model of a SES, a conceptual foundation is required from which the rules, properties, and agent tasks of the model can be derived. Without such a foundation, the intended model would need to be designed freely and based on intuition.

2. **Theories of human behaviour:** This exercise is aimed at modelling human behaviour in a theory-driven fashion. Hence, a set of theories of human behaviour needs to be identified which is suitable for describing the behaviours that agents need to engage in in the model.

3. **Theory formalisation:** As mentioned in Chapter 2, the theories of human behaviour need to be formalised in order to enable their implementation within an agent's behavioural routine. Without a formalisation, the elaborations made in most theories of human behaviour are expected to be too vague and informal for a modeller to be able to derive meaningful guidelines for the design and implementation of agents from them.

4. **Application of the HuB-CC framework:** One of the goals of this work is to use a conceptual framework for modelling human behaviour. The HuB-CC framework satisfies this requirement. Its stratification of the human cognition and decision-making process into elements as well as the mapping of theories to those elements needs to be used for theory selection as well as to inform the agents' internal structure.[1]

[1] A more detailed description of the framework is given later in the thesis.

5. **Comparability of theory implementations:** One of the motivations of implementing theories of human behaviour in agents is to be able to compare the behaviours of agents as a result of the theories. Therefore, the model needs to be developed in such a way as to enable an exchange of theories that guide the agents' decision-making and behaviour.

6. **Interdisciplinarity:** The conceptual part of this thesis is carried out with the help of domain experts. Therefore, the resulting model and theory implementations are required to be satisfactory from a domain expert's point of view.

The satisfaction of these requirements is tracked in the remainder of the written thesis.

4 Methods and Materials

The requirements formulated in Chapter 3 will be carried forward through the remaining chapters of this thesis. In this chapter, an overview of the methods and materials used for the subsequent design and implementation of the model is given. First, the MARS framework is described to illustrate the technical aspects of this work. This is followed by a description of the HuB-CC framework, which provides the conceptual frame within which the agents are designed. Next, the conceptual basis of the model is outlined, followed by the selection of theories of human behaviour.

4.1 The MARS Modelling Framework

The MARS modelling framework is a research project that is under ongoing development by the MARS Group at the University of Applied Science Hamburg (HAW Hamburg). Made up primarily of computer science students, the group is interested in bringing the tools and methods of computer science to other disciplines to create ABMs in interdisciplinary settings. The key motivation lies in creating synergistic effects, allowing researchers from different domains to learn and benefit from each other's expertise. Previous models were set in the domain of ecology, urban planning, traffic modelling, and public health, among others.

The MARS system is written in C Sharp (C#), a multi-paradigm programming language that was chosen by the MARS Group primarily to facilitate a hierarchical inheritance structure – as is common in OOP – of various abstract and non-abstract agent types and environment types. C# is fully supported by the cross-platform software framework .NET. As such, models that are written in MARS can be developed and executed on all common operating systems. The current version of the MARS runtime system is available in the form of a NuGet package. Installing the package in an Integrated Development Environment (IDE) – most notably Rider by JetBrains, which fully supports

.NET projects – enables developers to access its functionality and create their own models using C# and the available abstractions of agent types, entity types, and layer types. For further details, please see the MARS framework documentation.[1]

A typical ABM written in MARS features three key concepts. A brief overview of these concepts is offered below.

Agents An agent is an autonomous software instance and an active participant of an ABM simulation. In MARS, each instance of an agent type is executed by a thread. This enables the concurrent execution of behaviour routines of multiple agents. Unlike in sequential execution settings, this gives MARS agents the ability to interact with and influence their environment as well as each other simultaneously. The typical complications that can arise at and during runtime as a result of concurrency are handled by the MARS framework.

Entities An entity is any passive object that can be used in some way by an agent as a resource. Most recently, the MARS entity concept was employed in agent-based traffic models to represent vehicles, particularly cars and bicycles [41, 26, 27, 9].

Layers A layer is a section of a MARS model's environment. It usually encompasses objects and/or information of the environment that belong to the same class or category. Depending on the model, a layer can be spatially referenced. For instance, in an ecology model that involves trees, the environment's trees can be contained within one layer. Likewise, in a traffic model, a layer might comprise the environment's traffic network. In most cases, layers provide information for agents to interact with and structures for entities to situate themselves in. However, layers can also be without spatial reference. In that case, their functions tend to be administrative: spawning agents at the beginning of a simulation, holding resources for agents during a simulation, and de-spawning agents at the end of a simulation.

The MARS layer concept has been the focus of several recent research projects and is therefore particularly advanced. Spatially referenced layer types can integrate Geographic Information System (GIS) data that enrich a model with real-world geo-referenced information [19]. Provided that the pertinent data are available in a workable form, they

[1] https://mars.haw-hamburg.de/

can be used to develop ABMs with real-world reference frames. This can be especially promising for studying the behaviour of individuals in a specific environment. In comparison, the agent concept in MARS appears to be less sophisticated. At the technical level, the functionality required to develop reliable and performant ABMs is given. However, the internal structure of agents has room for improvement and expansion. It stands to reason that modelling the internal structure of an agent – that is usually intended to represent a living being – based on expertise from domains that study such structures can provide benefits to the expressiveness and efficacy of ABMs in MARS.

4.2 The HuB-CC Framework

The HuB-CC framework [10] is a conceptual tool that promotes theory-driven behaviour modelling. It is based on the Modelling Human Behaviour (MoHub) framework [33], which was developed by sustainability researchers for integration in natural resource use models – ABMs, among others – that aim to better understand the impact of human behaviour on meeting sustainability goals. The HuB-CC framework was designed by an interdisciplinary team of psychologists and sustainability researchers to systematize and inform modelling processes and to aid experimental and empirical researchers of SES in the field. To do so, the framework combines two components – **elements** and **theories** – that are described briefly in the following subsections.

HuB-CC Elements The HuB-CC framework features a set of elements that comprise the internal components of a human actor's decision-making process – from the perception and interpretation of external stimuli to the choice of a specific behaviour which is expressed to the environment. The elements are Perception, Attention, Learning and Updating, External Information Search, Memory Search, Valuation, Choice, Behaviour, Stable Characteristics, Situational Characteristics, and the Social and Biophysical Environment. While a detailed elaboration on all elements is beyond the scope of this work, a brief description of each can be found in the glossary. In Figure 4.1[2], a schematic representation of the elements is shown.

[2] In the original diagram, there is an undirected association between Valuation to Choice. For this thesis, it was replaced with an arrow. This is to indicate that, in the implemented model, the relationship between Valuation and Choice is strictly feed-forward.

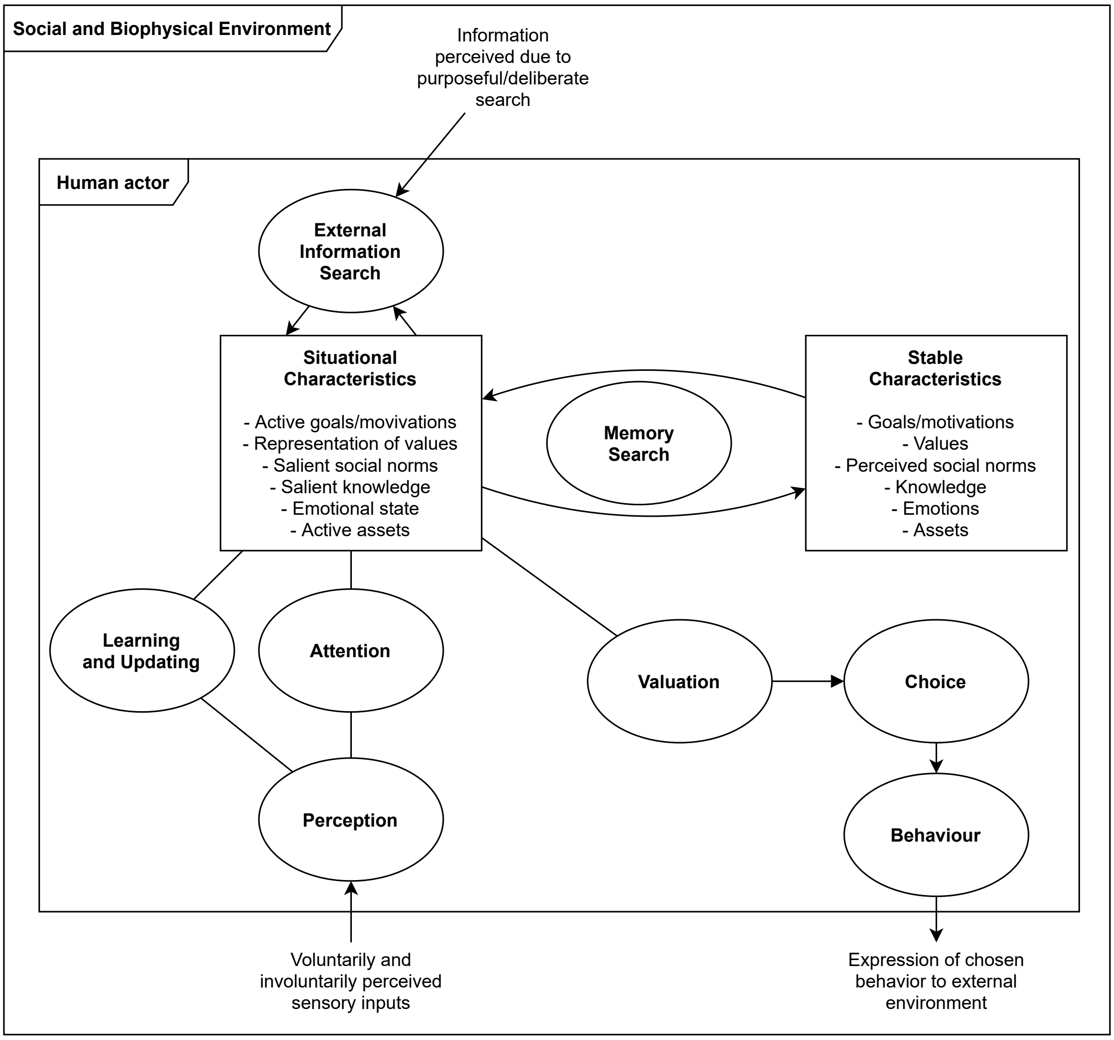

Figure 4.1: A schematic illustration of the elements of the HuB-CC framework (adapted from [10]).

Each element serves a specific role in the overall cognitive process. Based on its role, an element is shown to be associated with other elements. For example, the element Valuation is the process is evaluating a range of options to identify their desirability [10]. Hence, its output – a range of preferences, each corresponding to an option – is the input of the element Choice, which chooses an option from a range of possibilities under certain criteria [10]. In general, the framework indicates a division of labour and responsibilities among the elements as well as some degree of sequentiality in the cognitive processes. Furthermore, it should be noted that the human actor shown in Figure 4.1 is embedded in a Social and Biophysical Environment along with many other human actors. Actors form implicit and explicit connections via the elements Perception, External Information Search, and Behaviour. It is conceivable that a set of agents in an ABM might be made up in a similar fashion.

Theories of Behaviour The authors of the HuB-CC framework have aggregated 31 theories of human behaviour. Each theory has been mapped to the HuB-CC elements in an effort to describe the theory's scope and focus. For example, a theory can be mapped to Valuation and Choice. This might inform a modeller that the theory could serve useful in devising a formalization for agents of an ABM that, by design, is particularly interested in studying the agents' evaluation and choice process. Likewise, a theory that maps to Memory Search could help design agents with search and lookup mechanisms that are similar to those of humans – including a way to account for the occasional forgetfulness and inaccuracies that are typically foreign to computers.

While the examples for applications of the framework given in the paper are limited in context to natural resource use, the authors state that the framework is expressive enough to be applicable in a wide range of settings. In general, both the HuB-CC framework and the MoHub framework aim to bridge the language gap and knowledge gap between domain experts and behaviour modellers. Using frameworks such as MoHub and HuB-CC, the insights of such theories can be applied by modellers to a given behavioural model to inform the design and structure of the model's agents. The authors offer two general use cases for in which modellers or researchers might benefit from applying the framework. Both use cases start with an observation of some behaviour of interest. A brief description of each use case follows.

- **HuB-CC Use Case 1:** The modeller or researcher might be aware of – and possibly understand at an intuitive level – some of the aspects (elements) of the decision-making processes that result in the observed behaviour. But the theoretical underpinnings that would explain the behaviour at a formal level are missing. But such a level of understanding is often required to design robust behavioural models. The modeller can consult the HuB-CC theory map to identify theories that – based on the HuB-CC elements that map to them – might explain the behaviour of interest.
- **HuB-CC Use Case 2:** Rather than being aware of potential underlying processes that might explain the observed behaviour, the modeller might be aware of a few

candidate theories that might explain it. In this case, the modeller can use the HuB-CC theory map as a means to compare theories based on the HuB-CC elements that map to them. Such a comparison might help narrow down the theory selection process and enable the modeller to make a more informed decision.

4.3 Model Basis

In this section, the conceptual basis of the model that is designed for this thesis is described. First, however, the term Common-Pool Resource (CPR) game needs to be defined. A CPR game is a turn-based game in which, based on a set of rules, each player can extract a resource (usually a natural resource like wood, fish, etc.) from a common environment. The resource has some regrowth or reproduction rate, enabling the players to extract it over a number of rounds. Typically, the goal of such a game is twofold. On the one hand, each player aims to extract as much of the resource as possible for personal profit. On the other hand, the community must prevent the resource from becoming fully depleted. Therefore, the players need to to strike a balance between individual and collective interests.

Game Description

The model implemented for this thesis is a recreation of a CPR game that was played during a field study of a Colombian fisher community [31]. This serves as the model basis and addresses one of the requirements stated in Chapter 3. As described in the paper, the game requires fishers to extract fish from a common pool, similarly to the notion described in the previous paragraph. Since the game's core mechanic involves natural resources extraction, the behaviours displayed by its players are expected to be well-suited for being modelled with the HuB-CC framework (see Section 4.2). A game involves a group of four fishers and lasts 16 rounds. Each round represents a day of fishing and consists of the following sequential routine:

1. **Information:** The fishers are told the current fish population size and reproduction rate.
2. **Communication and Deliberation:** The fishers have some time to consider possible actions and/or talk to each other. They may coordinate with each other,

devise cooperative strategies, and even disclose their intended fishing goals to one another. They may also choose not to communicate with each other.

3. **Decision:** Each fisher individually and privately makes a final decision on her fishing goal. The goal can be freely defined and is not constrained by any rules or limitations. The goal is never disclosed to the other fishers.

4. **Action:** If enough fish are currently in the common pool to satisfy the fishing goals of all fishers, then that number of fish is extracted from the common pool and distributed accordingly. Otherwise, the available number of fish is distributed among the fishers proportionally to their fishing goals.

At the beginning of round 1, the fish population size is 50. Figure 4.2 shows the reproduction dynamics of the fish population.[3]

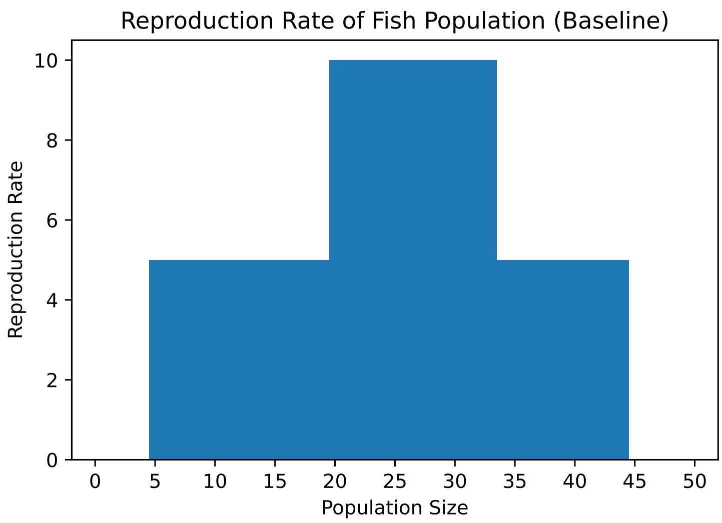

Figure 4.2: Baseline reproduction dynamics of fish population (adapted from [31]).

The game can be played in four variations, each representing a treatment in the study. In all variations, the fish population size is restored to 50 at the beginning of round 7. Figure 4.2 shows the reproduction dynamics that apply throughout the **Control** treatment. Alternatively, there are three other treatments – **Threshold**, **Risk**, and **Uncertainty**. In these treatments, the baseline reproduction dynamics shown in Figure

[3]For details on the biological rationale for the rate distribution, please see [31].

4.2 apply from round 1 to round 7. However, from round 7 onward, a climate event can occur with some probability that depends on the treatment:

- **Threshold:** a climate event occurs at the beginning of round 7 with 100% probability.
- **Risk:** a climate event occurs at the beginning of any round between round 7 and round 16 with 50% probability per round.
- **Uncertainty:** a climate event occurs at the beginning of any round between round 7 and round 16 with a probability between 10%-90% that is recomputed per round.

Figure 4.3 shows the altered reproduction dynamics as a result of a climate event. The reproduction rate of the fish population decreases substantially when the population size is between 5 and 28. This increases the challenge of extracting fish while preventing extinction of the fish population. Once a climate event has occurred, its impact on the reproduction dynamics lasts for the remainder of the game.

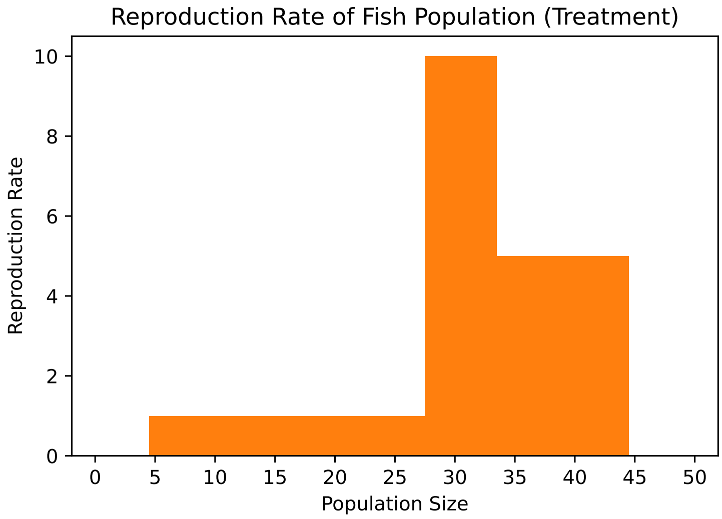

Figure 4.3: Treatment reproduction dynamics of fish population (adapted from [31]).

Player Knowledge

An important consideration when modelling such a game as an ABM is player knowledge. Specifically, it is important to track what is knowable to which player at what time.

Agents in an ABM can be modelled as omniscient beings. But an all-knowing agent would not make for an apt representation of a human. Therefore, constraining the knowledge base of agents can make for a more realistic model. In the model implemented for this thesis, the information in [31] about what is known and unknown to the players of the CPR game is used as a basis for constraining the agents' knowledge base. The following list contains a summary of that information:

- Players know the population size at all times.
- Players know the reproduction rate at all times.
- Players do not know each other's fishing goal decisions.
- Players do not know the number of rounds a game lasts.
- Players do not know which variation of the game they play.
- Players do not know the range of rounds during which a climate event can occur.

4.4 Behavioural Theories

To model the behaviour of players of the CPR game (see Section 4.3), the HuB-CC framework is consulted. Specifically, a small subset of the 31 available theories is chosen. This choice is made such that the subset of HuB-CC elements that the theories map to is considered to be able to adequately describe the main decision-making processes that are thought to take place while playing the CPR game. In this section, the chosen theories are described. They serve to address one of the requirements stated in Chapter 3.

Bounded Rationality The theory of Bounded Rationality is an extension of the theory of Rational Choice [35]. It was proposed by the social scientist Herbert A. Simon in 1957 [36]. Both theories describe human decision-making processes as being determined exclusively by the maximization of personal utility. The theory of Bounded Rationality additionally takes into account that human knowledge and capability for information processing is bounded (i.e., limited). On the one hand, there are intrinsic bounds such as the nature and computation power of the human brain. On the other hand, there are extrinsic bounds, such as aspects of the environment that are unknowable at the time of decision-making because the environment cannot provide them.

With respect to modelling human behaviour, Simon suggests a few of ways to "bound" an agent's decision-making processes [37]:

- Include a component of risk and/or uncertainty in the model
- Give an agent incomplete information about alternatives
- Add complexity to the model so great that the agent is unable to compute an ideal course of action deterministically and is instead forced to approximate it stochastically

In the context of Bounded Rationality, the act of Satisficing is often mentioned: foregoing a choice that might yield a high short-term reward in favour of a potentially higher long-term reward. This kind of decision-making usually takes hold when the decision-maker has incomplete information. The term aptly captures the dilemma that is prevalent in CPR scenarios: when the amount of resources is low, the extraction behaviour must be more conservative than when resources are plentiful. On the one hand, this implies a smaller short-term profit. On the other hand, it reduces the probability of complete depletion, thereby increasing the probability of higher long-term profit. According to Bounded Rationality, Satisficing is often paired with optimizing as the Satisficing human still seeks to maximize personal utility, but does so while considering external factors.

Table 4.1 shows the mapping devised by the HuB-CC framework between the theory of Bounded Rationality and the HuB-CC elements.

Table 4.1: Mapping between theory of Bounded Rationality and HuB-CC elements (adapted from [10])

Theory	Perception	Attention	Learning and Updating	External Information Search	Memory Search	Valuation	Choice	Behaviour	Stable Characteristics	Situational Characteristics	Social and Biophysical Environment
Bounded Rationality						x	x				

The theory of Bounded Rationality maps to the elements Valuation and Choice. This is because it proposes a model only for how humans evaluate potential actions under given circumstances and for how an action is chosen. The theory does not make descriptions of other aspects of the human decision-making process (as laid out by the elements of the HuB-CC framework). While it does mention, for example, the potential influence of environmental factors on the decision-making process (which might fall under Social and Biophysical Environment), it does not elaborate on these factors in a way that might inform the description of the environment of an ABM in which agents are required to work with incomplete information. The theory is considered to be well-suited for modelling the chosen CPR game (see Section 4.3) because its players are required to maximize utility while preventing resource extinction while given incomplete information.

Trust and Reciprocity The theory of Trust and Reciprocity [4] proposes a model for how the two traits trust and reciprocity factor into a one-time, two-way transaction between two individuals who do not know each other (i.e., are interacting with each other for the first time). The authors observe that many economic theories assume that individuals act in their own self-interest – in line with the theory of Rational Choice. In such theories, a deviation from behaviour that is not in the actor's self-interest is viewed as irrational. However, in group settings, there exist situations where self-interested behaviour impacts all group members – even the actor – negatively, implying that irrational behaviours might sometimes be advantageous. Since individuals who exhibit such behaviours rely on the notion that a foregone short-term benefit will be compensated by a long-term benefit (a reciprocation), they require a degree of trust in other participants of the transaction.

The authors propose that trust is a so-called economic primitive: a trait that can be expressed by participants of an economic transaction regardless of previous experience with and even knowledge about the other participants.[4] The transaction is formally described as follows. A trustor X performs an action by which she places a trust in a trustee Y, who can break the trust by not reciprocating the action or keep the trust by reciprocating the action. In other words, X puts herself at risk by giving Y the right to make a decision, and Y makes a decision that affects both X and Y. In summary, trust is the act of exposing oneself in a transaction to the consequences of actions of the other participant. Reciprocity is the act of choosing an action that benefits both participants of the transaction.

Table 4.2 shows the mapping devised by the HuB-CC framework between the theory of Trust and Reciprocity and the HuB-CC elements.

Table 4.2: Mapping between theory of Trust and Reciprocity and HuB-CC elements (adapted from [10])

Theory	Perception	Attention	Learning and Updating	External Information Search	Memory Search	Valuation	Choice	Behaviour	Stable Characteristics	Situational Characteristics	Social and Biophysical Environment
Trust and Reciprocity	x					x	x		x		x

[4]The term "primitive" is reminiscent of primitive data types in computer science. These are built-in, low-level data types and serve as basic building blocks of a programming language. Analogously, the suggestion that trust is an economic primitive implies that it is a core trait of humans that is intrinsically present and does not first need to be assembled/composed by other primitives.

The theory maps to the elements Perception, Valuation, Choice, Stable Characteristics, and Social and Biophysical Environment. The elements Perception and Social and Biophysical Environment are included because the theory describes a transaction, i.e., a process which involves the intake of information (Perception) from another individual in the (social) environment. Since X and Y each have to make a choice between two options – to trust or not to trust and to reciprocate or not reciprocate, respectively – they must also evaluate the desirability of those choices. This accounts for Valuation and Choice, respectively. Finally, the theory considers trust to be an economic primitive, making it a trait that is permanent and not prone to (spontaneous) change. This accounts for the element Stable Characteristics.

In the CPR game (see Section 4.3), players can choose to communicate or not communicate with each other. The act of communicating can be viewed as placing a trust in the recipient of the communicated information. Likewise, players can choose to adhere to the fishing goal that was disclosed during a communication or to change their mind in private and deviate from that disclosed goal. The act of adhering to the disclosed fishing goal (i.e., the act of not lying) can be viewed as an act of reciprocity. It contributes an act of honesty to the community, thereby preserving trust within the community.

Social Norms The theory of Social Norms [6, 8] is a theory that establishes a connection between norms at the societal level and expectations at the individual level. A social norm is defined as a rule of behaviour such that individuals prefer to conform to it [5]. This preference is conditioned on two types of social expectation. The first type is called **empirical expectation**, which is an individual's belief that most people in her network conform to the rule of behaviour. The second type is **normative expectation**: an individual's belief that most people in her network believe that she should conform to the rule of behaviour. According to the theory of Social Norms, both types of expectation must be high in order for the individual's preference to be sufficiently high to conform to a given social norm.

According to Cristina Bicchieri, the author of this theory, social norms have been studied extensively and in many different contexts. But the definitions used to describe social norms are often not operational (i.e., not measurable). One of the key motivations in developing the theory of Social Norms was to be able to identify social norms in a given community and measure the extent to which they are followed by members of that community (for example, see [7]). Since the theory defines social norms to be closely linked to social expectations, it proposes that collective expectations of a community can be moderated by tapping into the social norms of the community. Expectations, in turn, are linked closely to behaviour. Therefore, in summary, the theory of Social Norms is used to identify norms and the expectations that community members associate with them in order to change individual behaviour in some desirable manner. Applications of the theory have shown empirically that changing either empirical expectations or normative expectations or both at a sufficiently large scale results in changes in collective behaviour [8].

Table 4.3 shows the mapping devised by the HuB-CC framework between the theory of Social Norms and the HuB-CC elements.

Table 4.3: Mapping between theory of Social Norms and HuB-CC elements (adapted from [10])

Theory	Perception	Attention	Learning and Updating	External Information Search	Memory Search	Valuation	Choice	Behaviour	Stable Characteristics	Situational Characteristics	Social and Biophysical Environment
Social Norms	x					x	x		x	x	x

This theory maps to the same elements as the theory of Trust and Reciprocity does, with the additional element of Situational Characteristics. This is because empirical expectations and normative expectations are said to be dynamic traits that can change readily over time, depending on societal trends that an individual perceives. The HuB-CC framework distinguishes between stable representations of social norms – which are part of an individual's Stable Characteristics and are relatively persistent over time – and salient representations of social norms – which are influenced by current circumstances and usually a derivation of the stable representation.

The field study described in Section 4.3 [31] played the CPR game with members of a Colombian fisher community. It stands to reason that this community has a set of social norms that are based on their craft, trade, and other local and social factors. However, the paper does not describe such aspects of the community, likely because the study was not focused on them. It appears that the community was approached to play the CPR game more for their domain expertise (fishing) than for their community's attributes.

5 Design and Implementation

After identifying the requirements (see Chapter 3) and gathering and understanding the necessary methods and materials (see Chapter 4) for this project, the intended model can be described. This chapter begins with a structural description of the model, covering the classes and components that comprise it. This is followed by formalisations of the theories described in Section 4.4. The formalisations form the basis of the agents' internal structure and behaviour routine, which is outlined in the subsequent two sections. Lastly, some additional implementation details and considerations are touched on.

5.1 Model Description

In this section, the structure of the model that was implemented for this thesis is described. The description is organised in a top-down fashion, first highlighting the broad and general structural model components before gradually honing in on pertinent details.

Layer Types

Figure 5.1 shows the environment of the model.

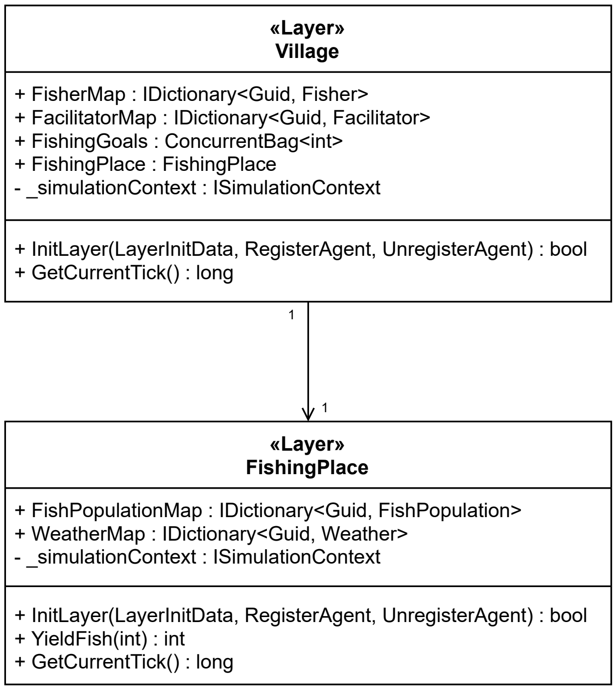

Figure 5.1: Class diagram showing the layer types of the model

The environment consists of two layer types: `Village` and `FishingPlace`. The `Village` represents the place at which the `Fisher` agents live. Accordingly, it has a property `FisherMap` in which references to the `Fisher` agents are held throughout the simulation. Furthermore, the layer type enables communication between the `Fisher` agents via the property `FishingGoals`. The object of type `ConcurrentBag` enables the concurrently running agent threads (see Section 4.1 for details) to deposit and get messages without the simulation encountering concurrency issues. The agent type `Facilitator` is responsible for clearing the contents of `FishingGoals` in between rounds so that it is ready for the next group communication. Furthermore, the `Village` grants the `Fisher` agents access to the layer type `FishingPlace`. The agent type `FishPopulation` lives on this layer and a reference to it is held in the property `FishPopulationMap`. `Fisher` agents can access the `FishPopulation` via the method `YieldFish(int)` to extract a desired number of fish. The agent type `Weather` also lives on this layer and can potentially impact the fish population (as described in Section 4.3).

A simulation of the model features exactly one layer instance of each of the two layer types. The directed association shown in Figure 5.1 indicates that the `Village` has a reference to the `FishingPlace`, but not the other way around. The private property `_simulationContext` as well as the methods `InitLayer` and `GetCurrentTick` are implemented based on the layer types' connections to MARS interfaces. A class diagram that illustrates this connection is included at the end of this section.

Layer Types and Agents Types

To expand on the initial view of the model environment, the agent types are extracted from the layer types in Figure 5.2 and shown in individual classifiers. For clarity and legibility, some of the properties and methods of the layer types that are included in Figure 5.1 are omitted in this diagram.

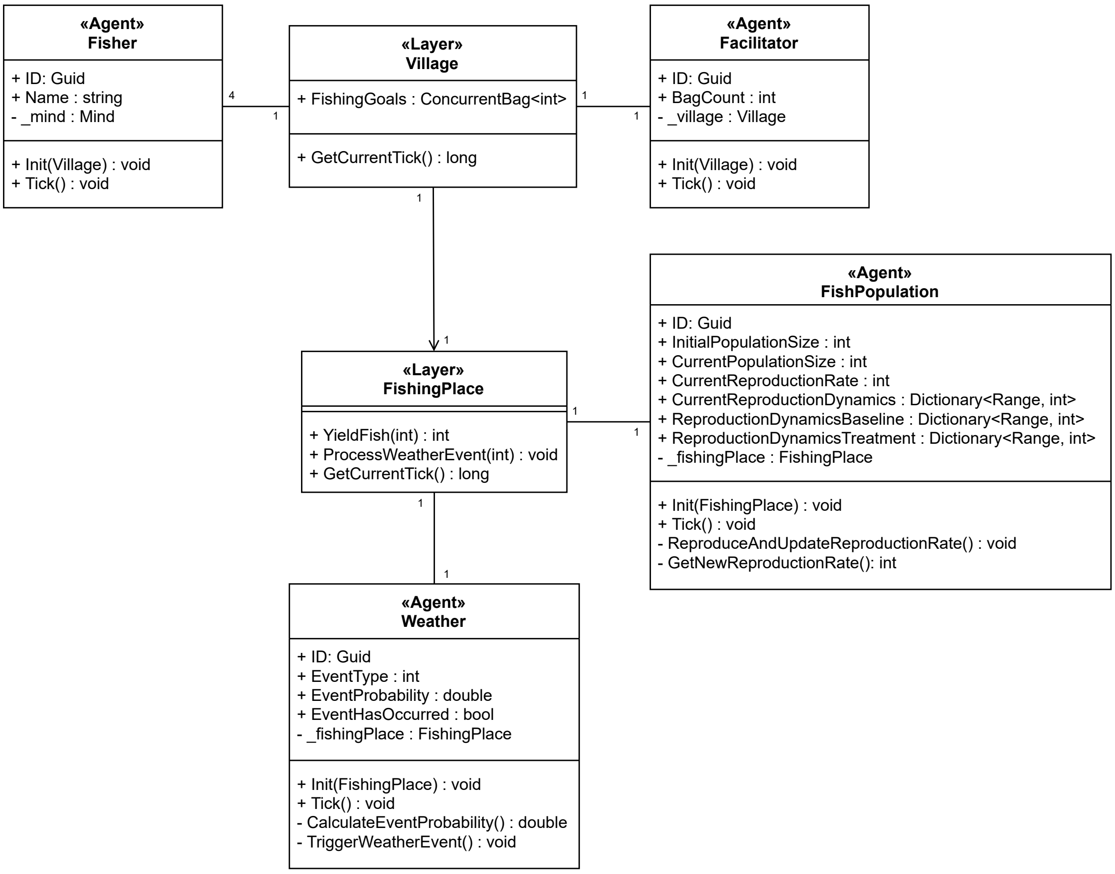

Figure 5.2: Class diagram showing the layer types and agent types of the model

A `Fisher` has a public property `Name` and a private property `_mind` of type `Mind`. `Name` is included for easier identification of agents in the output file during manual review of simulation results. The `Mind` houses the agent's decision-making structure and logic. As mentioned in the description of Figure 5.1, the property `FishingGoals` in the layer type `Village` is used for communication among the `Fisher` agents. The `Facilitator` clears the contents of this collection at the end of each round and tracks its size in the property `BagCount` for verification purposes in the simulation output file.

The agent type `FishPopulation` has properties to track the fish population's size (`InitialFishPopulationSize` and `CurrentFishPopulationSize`) and reproduction rate (`CurrentReproductionRate`). The reproduction dynamics are stored in a `Dictionary<Range, int>`. In this dictionary, a key is a population size range within which a reproduction rate is defined, and its value is that reproduction rate (see Figure 4.2 and Figure 4.3 for diagrams showing the ranges and rates defined in [31]). The `FishPopulation` has a method `ReproduceAndUpdateReproductionRate()`, which increases its `PopulationSize` by the value determined via the current reproduction dynamics. As a result of the change in population size, the reproduction rate might change, which is adjusted by the method `GetNewReproductionDynamics()`.

The agent type `Weather` has a property `EventType`, based on which an `EventProbability` is determined. The value of `EventType` can be 0, 1, or 2, corresponding to the three treatments that involve potential climate events (see Section 4.3). The property `EventHasOccurred` tracks whether an event has occurred to assure that a climate event occurs only once per simulation (as defined by the rules of the CPR game). For the treatment **Risk**, the method `CalculateEventProbability()` determines a new probability once per round for a climate event to occur. If a climate event occurs, it is triggered via `TriggerClimateEvent()`.

A simulation of the model features exactly four instances of the agent type `Fisher`, one instance of the agent type `FishPopulation`[1], one instance of the agent type `Facilitator`, and one instance of the agent type `Weather`. All associations between agent types and their corresponding layer types are undirected because the agent type holds a reference to the layer type it lives on and, likewise, the layer type holds a reference to

[1] The fish population is modelled as a single agent for simplicity and because the CPR game explicitly does not distinguish between different kinds of fish [31].

the agent types it has spawned in the corresponding property *Map (see Figure 5.1 for the names of these properties).

Note: The functionality of the Weather agent is fully implemented and the model is capable of processing climate events and adjusting the reproduction dynamics of the FishPopulation accordingly. However, during the development of the model, it was found that the theories selected to model the Fisher agents' behaviour do not offer insights into how humans behave as a result of incurring an uncertainty or a risk. Therefore, to avoid speculative behaviour modelling and to adhere to what is given by the theories, no behaviour related to a change in reproduction dynamics is currently implemented in the Fisher agents. Consequently, the simulations that were run for this thesis also do not include the three corresponding treatments (see Section 4.3). This feature can, however, be activated in the model by uncommenting a few lines of code and creating a connection from the Weather agent to the Fisher agent via the layer types (removing the directed association between Village and FishingPlace shown in Figure 5.1).

Connection to the MARS system

To conclude the description of the model's structure, Figure 5.3 shows the connection of the agent types and layer types to the MARS system.

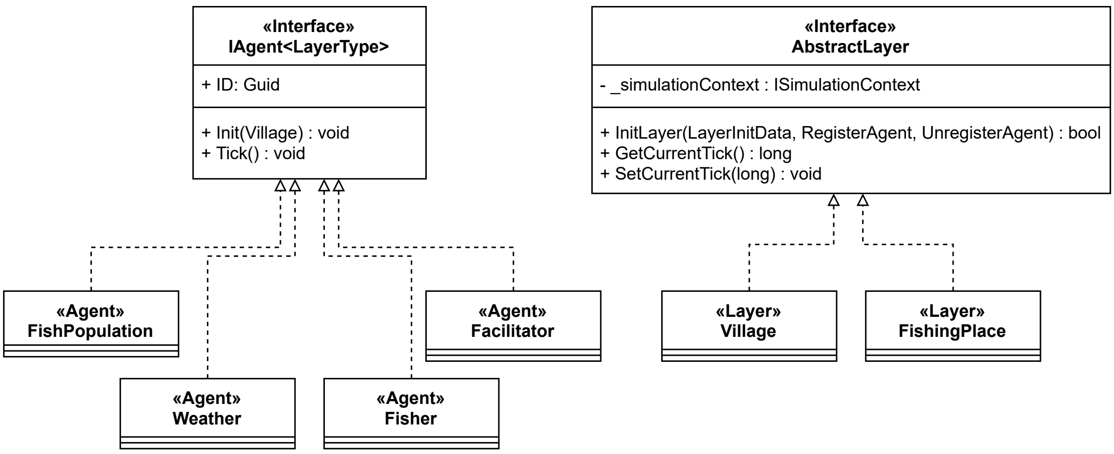

Figure 5.3: Class diagram showing the MARS interfaces implemented by the agent types and layer types of the model

For simplicity, all properties and methods of the implemented agent types and layer type are omitted in this diagram. The properties and methods required by the interfaces

`IAgent` and `AbstractLayer` provide the technical capabilities necessary to develop and execute the ABM. The agent type property `ID` acts as a unique identifier, while the method `Init(LayerType)` receives the layer type on which the agent lives and the method `Tick` performs a behavioural routine of the agent during each time step (hereafter referred to as tick) of the simulation. The layer type property `_simulationContext` is a reference to an object of type `ISimulationContext` that contains information about the simulation during its execution time (e.g., the current simulation tick, which can be requested by agent types by calling the layer type method `GetCurrentTick()`). Finally, the layer type method `InitLayer()` (the three formal parameters are omitted here for legibility) is responsible for performing administrative tasks at the beginning of the simulation, such as registering and spawning agent instances. (The layer type method `SetCurrentTick(long)` is implemented as requested by the interface's contract; but it is not used in the model).

5.2 Theory Formalisations

In this section, the formalisations derived from the behavioural theories that were described in Section 4.4 are outlined. With the model structure (see Section 5.1) and the formalisations of the theories, the `Mind` object that was mentioned in the previous section as well as the behavioural routine of the `Fisher` agents can be described. These formalisation aim to address one of the requirements stated in Chapter 3.

Bounded Rationality By design, the CPR game on which the model implemented for this thesis is based forces the agents to act rationally within pre-defined and omnipresent bounds. Each agent's primary goal is to maximize utility, making their behaviour inherently rational per definition. In addition, based on the information obtained from [31] about the players' knowledge base (see Section 4.3), each agent's knowledge base is always incomplete. This introduces bounds to their decision-making processes, as defined by Simon (see Section 4.4). Since the constraints on knowledge are implemented in the model as they were enforced in the real-world CPR game, the agents can be considered boundedly rational actors.

Trust and Reciprocity As outlined in Section 4.4, both trust and reciprocity are Stable Characteristics. According to the HuB-CC framework, this means that their values are relatively consistent over time. Since the time frame of the implemented model is relatively short (16 rounds), a `Fisher` agent's trust and reciprocity is implemented each as a `double` between 0.0 and 1.0 that is constant throughout the simulation. These values can be inserted into the simulation via the agent initialization file[2]. Trust and reciprocity are positively correlated [2, 15]. Therefore, sensible values for these two properties are likely to be close to each other.

As outlined in Section 4.4, trust is an act of exposing oneself to the consequences of a communication partner's actions. Reciprocity is the act of that communication partner such that both participants of the communication benefit. During the phase **Communication and Deliberation** of the CPR game (see Section 4.3), a `Fisher` can choose to communicate or not communicate. The former corresponds to an action rooted in trust, whereas the latter corresponds to an action rooted in lack of trust. Likewise, during the phase **Decision**, a `Fisher` can choose to adhere to the fishing goal disclosed to the other `Fisher` agents or to increase it. The former corresponds to an act of keeping the trust (i.e., an action rooted in reciprocity), whereas the latter corresponds to an act of breaking the trust (i.e., an action rooted in lack of reciprocity). Telling the truth benefits the community, whereas not telling the truth (by increasing the fishing goal) benefits only the individual.

Based on the rules of the CPR game (see Section 4.3), a `Fisher` agent has enough information in its knowledge base to determine whether another agent changed its fishing goal during the previous round. The only requirement for this is that all four agents chose to communicate during the previous round. In that case, a `Fisher` has enough information to perform the following calculation. For this purpose, let r be the current round, S_{r-1} the population size during the previous round, $G = \{g_1, g_2, g_3, g_4\}$ a set of four distinct fishing goals g_i (one reported by each `Fisher` agent), R_r the current reproduction rate, and S_r the current population size.

$$S_{r-1} - \sum_{1}^{|G|} g_i + R_r = S_r \tag{5.1}$$

If this equation is false, then at least one `Fisher` agent did not adhere to its communicated fishing goal during the previous round. However, the `Fisher` that performs the calculation cannot know which agent did so because, based on the rules of the CPR

[2]This is a Comma-Separated Values (CSV) file that lists initial values of some of the agent's properties.

game, agents cannot disclose their decided fishing goals to each other (see Section 4.3). Therefore, the consequence of discovering that a community member lied cannot be directed at that community member. It can only be directed at the community in general, if at all.

The theory of Trust and Reciprocity does not describe how an individual responds to having her trust broken. This is because the theory limits itself to one-time transactions between strangers (see Section 4.4). Therefore, a `Fisher` agent's response to discovering that someone in the community did not reciprocate is not modelled after the theory. To moderate the influence of the constant properties `Trust` and `Reciprocity` on an agent's future choices, an additional property `Scepticism` (a `double` between 0.0 and 1.0 that can change during the simulation) is introduced. When a `Fisher` finds that someone in the community lied, its `Scepticism` value is likely to increase, representing an increase in scepticism towards the community. The higher a `Fisher` agent's value of `Scepticism`, the lower the probability of that agent choosing actions that are rooted in trust or reciprocity during future rounds.

Social Norms As described in Section 4.4, the theory of Social Norms focuses on identifying and measuring norms in communities and leveraging expectations of community members to enact positive behavioural changes. In other words, the substance of this theory is about changing individual behaviours at a large scale and over a long period of time. During the design and development process of this thesis, it was found that the theory is not readily applicable to the relatively small-scale and short-term setting given by the chosen CPR game. Given the constraints imposed by the game's setting and rules, no way was found to devise a formalisation for changes in expectations and social norms that can be readily integrated into the model.

Therefore, rather than focusing on changing social norms (as the theory of Social Norms does), the idea of an existing social norm that impacts behaviours of a community at a large scale was integrated into the model in the form of a hyperparameter. Since [31] provides no insights on what might be realistic social norms for the community that whose behaviour was studied in the paper, these norms are estimated and are therefore broad in nature. In the agent initialization file, the property `SocialNorm` can be set to 0, 1, or 2. The value must be the same for all four agents. The values have the following meaning:

- 0: Individualism: `Fisher` agents with this setting skip the phase **Communication and Deliberation** and do not engage with each other. Such a simulation essentially involves agents that act boundedly rational and without the influence of the formalisation devised for the theory of Trust and Reciprocity.
- 1: Collectivism with potential decay: `Fisher` agents behave as described in the preceding section on Trust and Reciprocity. They can communicate and reconsider their fishing goals, but they can also grow increasingly sceptical of each other. The initially collectivistic approach of the community might therefore decay over time, making the community members increasingly dishonest and focused exclusively on utility maximisation.
- 2: Collectivism and sustainability: `Fisher` agents with this setting engage in communication and devise a common fishing goal cooperatively. They do so by first choosing a fishing goal individually, and then forming the average of all four fishing goals. Furthermore, if the average (times four) is found to deplete more than a certain threshold of the current fish population, the community fishing goal is reduced by a certain factor, promoting a sustainable use of the common resource at the community level.

By being able to implicitly activate and deactivate theories, an attempt is made to make their individual impact on the agents' behaviour comparable. This addresses one of the requirements stated in Chapter 3.

5.3 Agent Structure: Fisher

Given the structure of the model (see Section 5.1) and the formalisations of the behavioural theories (see Section 5.2), the internal structure and behaviours of the `Fisher` agent can be explored in detail. This section serves that purpose. The connection between the `Fisher` and the `Mind` is described, followed by a detailed overview of the internal structure of the `Mind`. Figure 5.4 shows the `Fisher` agent's association with its `Mind`. The implementation of this structure aims to address one of the requirements stated in Chapter 3.

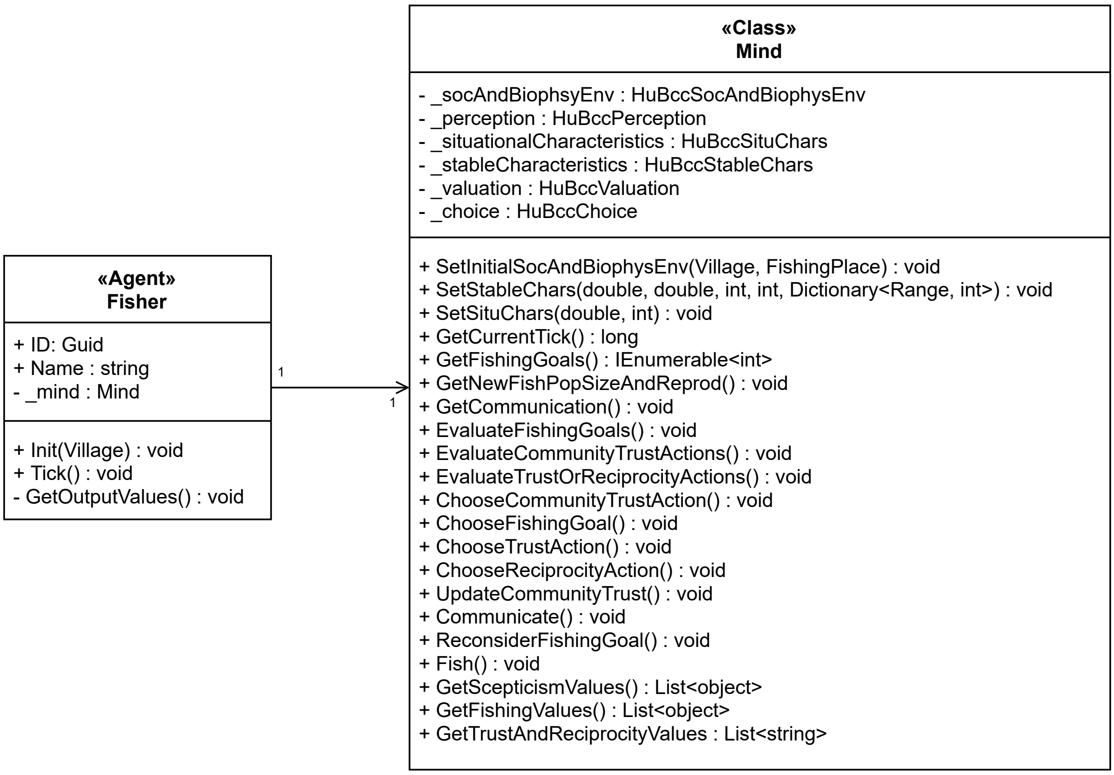

Figure 5.4: Class diagram showing association between `Fisher` and `Mind`

The `Mind` features a literal representation of the HuB-CC elements that the chosen theories (see Section 4.4) map to. Hence, the `Mind` provides the agent with dedicated components, each responsible for a specific task. The class `HuBccSocAndBiophysEnv` holds representations of the agent's knowledge that come directly from its environment. `HuBccPerception` is responsible for perceiving that information.[3] The class `HuBcc-SituChars` holds a set of situational characteristics, i.e., properties and information that tends to change over time.[4] Likewise, `HuBccStableChars` holds properties and values that are permanent. Finally, the classes `HuBccValuation` and `HuBccChoice` perform evaluation and decision tasks, respectively, on behalf of the `Fisher` agent.

The `Mind` encapsulates these classes from the `Fisher`, hence acting as a façade for the `Fisher` [18, p. 185–194]. Using the public methods the `Mind` makes available

[3]In this model, `HuBccPerception` also performs tasks that would be more accurately classified as tasks of an External Information Search component.

[4]In this model, `HuBccSituChars` also performs tasks that would be more accurately classified as tasks of a Memory Search component.

to the `Fisher`, the agent can make requests to its `Mind`. The processing of these requests is orchestrated by the `Mind` internally and without the agent being aware of the existence of the various internal components. Figure 5.5 shows the division of labour and responsibility among the internal components of the mind in greater detail. The rationale for this design choice is that a human also does not explicitly access parts of her brain in order to retrieve certain information or perform certain cognitive tasks. Rather, most of these processes happen intuitively and subconsciously, and typically, they begin with the mental formulation of an idea or some need. Similarly, a `Fisher` agent can make a request to its `Mind` and let it perform the required steps explicitly to complete the task.

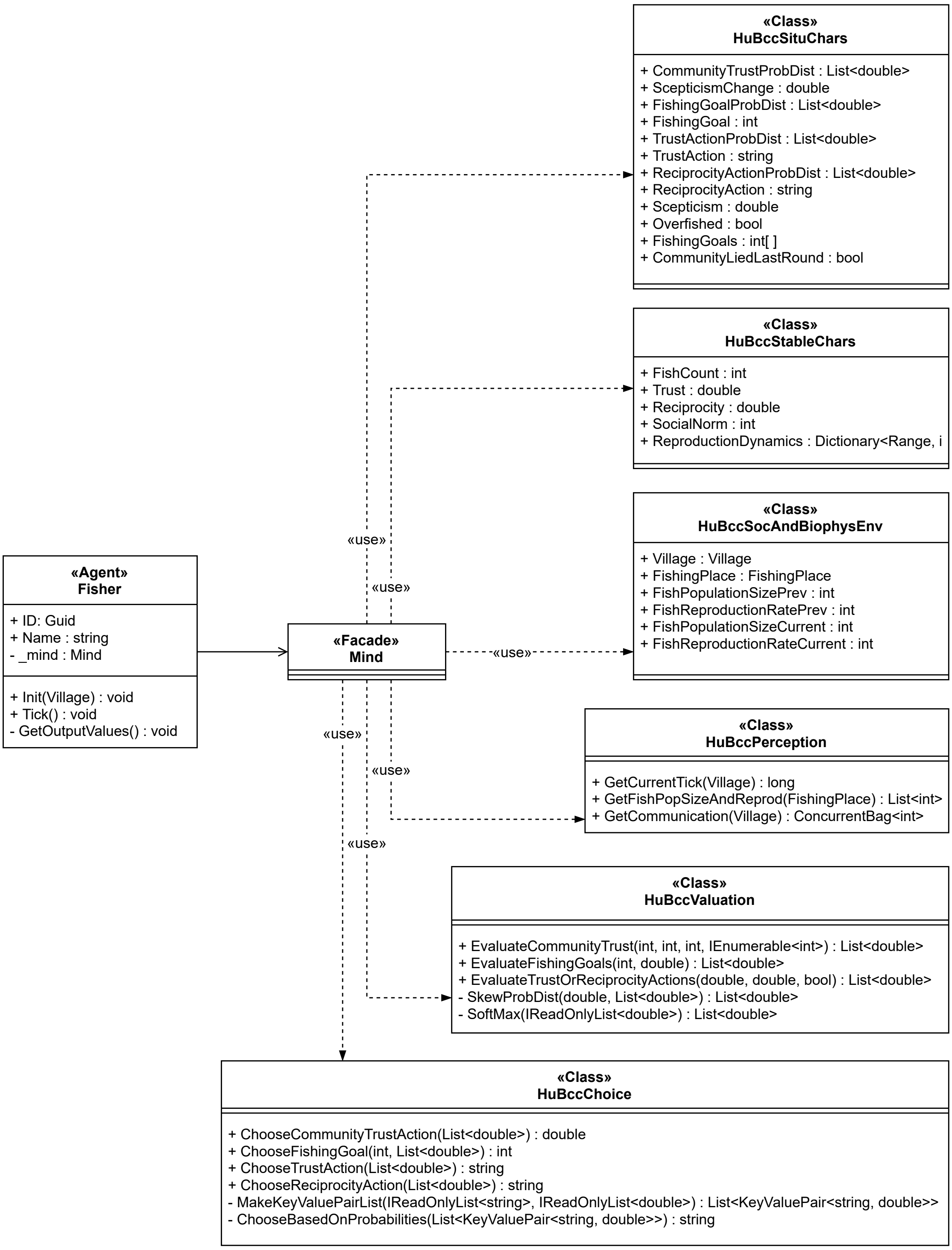

Figure 5.5: Class diagram showing the `Mind` and its internal components as a façade for the `Fisher`

5.4 Agent Behaviour

In the preceding sections, the model structure, theory formalisation, and agent structure were described. To complete the description of the model, the behavioural routines of the agents need to be outlined. This section offers an overview of the main processes of the agent types `Fisher`, `FishPopulation`, and `Facilitator`.

Routine of the `Fisher`

To model the behavioural routine of the fisher similarly to how the real-world CPR game (see Section 4.3) proceeded, it is not feasible to map one round of the game to one simulation tick. In that case, the agents would need to observe the environment, communicate with each other, evaluate and choose a fishing goal, and fish in one tick. Since MARS agents run concurrently (see Section 4.1), coordinating these tasks simultaneously for four agents would be challenging. Furthermore, it would not make for an accurate representation of the way the original CPR game proceeded from one phase to the next. Therefore, while designing and implementing the model, it was decided to sequentialise the routine of one round of the CPR game by splitting it up into a number of ticks. Figure 5.6 shows the resulting routine for the `Fisher` agent. Each round is divided into nine ticks. Since the CPR game consists of 16 rounds, a simulation consists of $9 \times 16 = 144$ ticks. Depending on which tick it is during a given round, only a certain part of the overall behaviour routine is completed. This is comparable to a stage-based game in which all players advance from one stage to the next only once every player has finished the current stage.

The routine features two Perception tasks, four Valuation-Choice tasks[5], and two Behaviour tasks. Other HuB-CC components within the `Mind` – that are not included in Figure 5.6 – are accessed during the execution of these tasks. A description of the task performed during each tick follows.

[5]In this model, Valuation and Choice always occur pairwise, with Choice receiving an input from Valuation and choosing an action. However, it is also conceivable for behavioural models to feature Valuation-Choice subroutines that involve a feedback loop from Choice back to Valuation to, for example, re-evaluate a set of options based on alternative criteria.

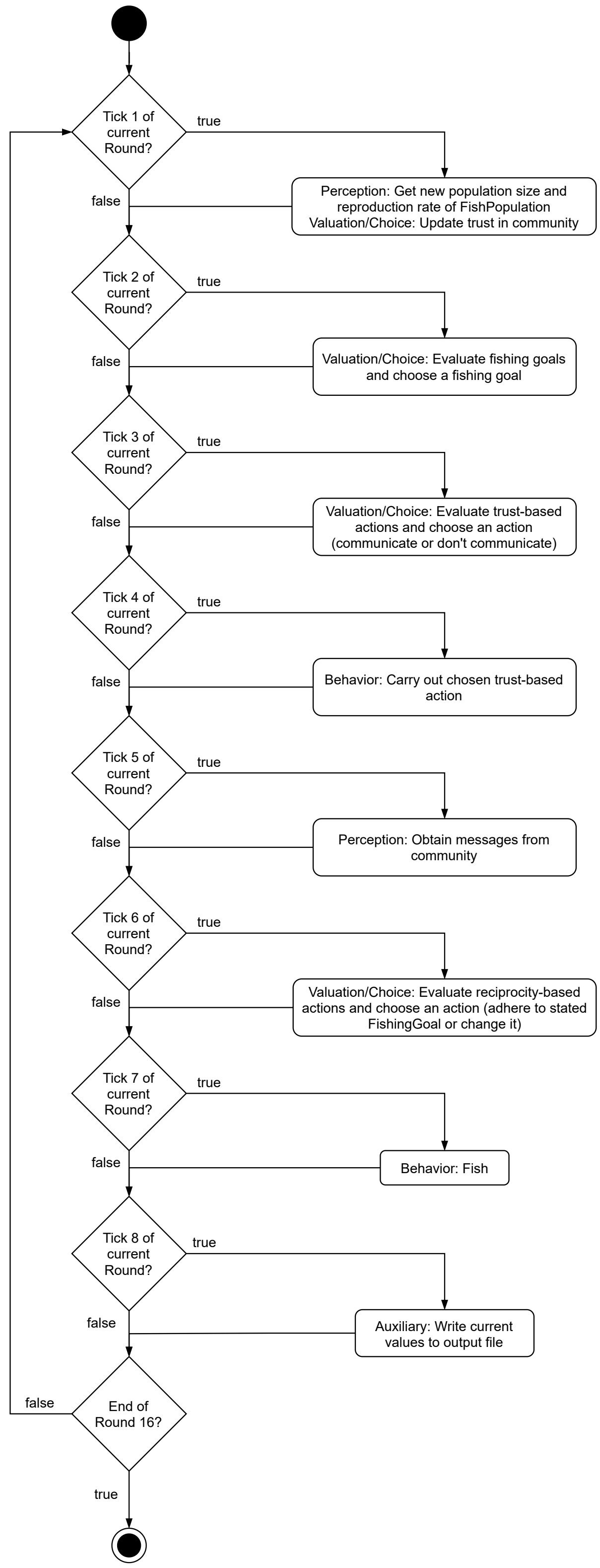

Figure 5.6: Activity diagram showing the behavioural routine of the `Fisher`

Tick 1 During this tick, each `Fisher` agent acquires updated information for the environment – specifically, the `CurrentPopulationSize` and `CurrentReproductionRate` of the `FishPopulation`. With the updated information, the agent can perform the calculation listed in Section 5.2 to evaluate (and potentially update) its trust in the community. A change in trust is reflected by an increase in the agent's `Scepticism`.

Tick 2 During this tick, each `Fisher` evaluates a range of possible fishing goals and chooses one of them. The decision is based on the `CurrentPopulationSize` as well as the agent's `Scepticism` – if applicable, based on the value of `SocialNorm` (see Section 5.2). In general, fishing behaviour is modelled as being relatively greedy when there are many fish and relatively conservative when there are few fish. When the agent's `Scepticism` in its community is high, this behaviour gets skewed towards greed even when the `CurrentPopulationSize` is low. The rationale for this is that such an agent is less concerned with cooperation within the community and more focused on maximising personal utility, since it assumes that its peers are doing the same – as indicated by their dishonesty during previous rounds.

Tick 3 During this tick, each `Fisher` evaluates whether or not to communicate with the community. This is a `Trust`-based action in the model (see Section 5.2 for the formalisation of the theory of Trust and Reciprocity). If `Scepticism` is high, it can skew the valuation and choice process towards not communicating, indicating the agent's lack of trust in the community. This trend can be further exacerbated if, during tick 1, it was found that a community member lied during the previous round.

Tick 4 During this tick, each `Fisher` performs the action chosen during tick 3. Specifically, the agent either reports or does not report the fishing goal chosen during tick 1 to the `Village`.

Tick 5 During this tick, each `Fisher` retrieves the fishing goals reported by the other agents from the collection in the `Village`.

Tick 6 During this tick, each `Fisher` evaluates whether to adhere to or change the fishing goal that was chosen during tick 2 and potentially reported to the community during tick 4. In the model, this is a `Reciprocity`-based action (see Section 5.2 for the formalisation of the theory of Trust and Reciprocity). The process of valuation and choice here works analogously to the one that is performed during tick 3. If a `Fisher` chooses to change its fishing goal, it is increased by some percentage before advancing to the next tick.

Tick 7 During this tick, each `Fisher` approaches the `FishingPlace` and aims to extract the number of fish it has chosen. A `Fisher` always catches as many fish as it has aimed to, unless there are not enough fish available to accommodate the chosen fishing goal.

Tick 8 During this tick, each `Fisher` obtains a set of property values from the `Mind` to write to the simulation output file. **Note:** this step is necessary for technical reasons. In devising the internal agent structure shown in Figure 5.4 and Figure 5.5, the properties of the agent were moved from the class `Fisher` into the `Mind` and its associated HuB-CC classes. The MARS system, however, can only write agent properties to output media if those properties are explicitly present in the agent type class. Therefore, at the end of each round, these values are retrieved from the `Mind` so that the MARS system can write them to an output file for later analysis of simulation results. While this is not ideal and caused many properties to be duplicated, it was deemed worthwhile for this thesis to enable the development of an internal agent structure.

Note: If integrated into the simulation flow, the `Weather` agent would become active during tick 0 of a given round.

Interaction between `Fisher` and `Mind`

Figure 5.7 shows an interaction sequence between the `Fisher` and its `Mind` during a Valuation process. As indicated in Figure 5.6, there are four such processes during each round. To illustrate the typical flow of these processes, the evaluation task `EvaluateFishingGoals()` (performed during tick 2 of each round) is taken as an example.

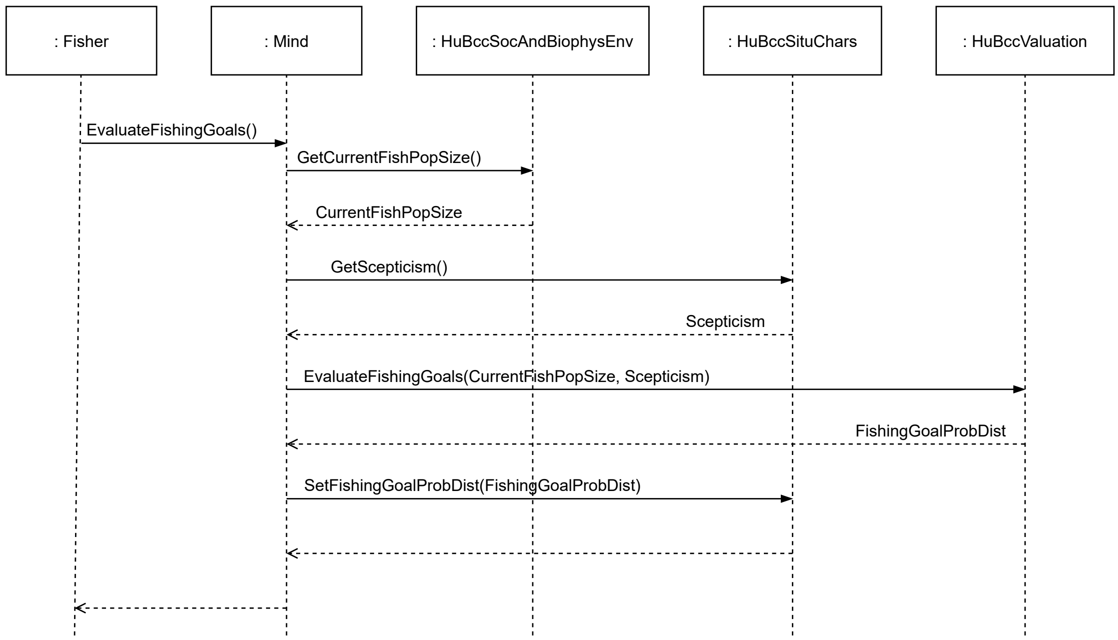

Figure 5.7: Sequence diagram showing interaction between `Fisher` and `Mind` (Example: `EvaluateFishingGoals()`)

The `Fisher` sends a request to its `Mind`, asking for an evaluation of fishing goals. The `Mind` proceeds to acquire the information necessary to perform the task. An evaluation of fishing goals requires the `CurrentFishPopulationSize`, which is obtained from the `HuBccSocAndBiophysEnv`. Furthermore, since the range of fishing goals is impacted by the `Fisher` agent's trust towards the community, the current value of the agent's `Scepticism` is retrieved from `HuBccSituChars`. The information is fed into `HuBccValuation`, where the valuation task is performed. The return value is stored in `HuBccSituChars` for subsequent use by `HuBccChoice`.

Valuation Process

Continuing to zoom in on the internal activities of the `Mind`, Figure 5.8 shows each step of the valuation task that is performed in `HuBccValuation`, as shown in the sequence illustrated in Figure 5.7. In the implemented model, a valuation process is generally carried out by devising a probability distribution over the possible actions. In the example `EvaluateFishingGoals()`, the valuation of fishing goals consists of devising probabilities for three possible fishing ranges: low, medium, high. The probabilities listed in

Figure 5.8 correspond to this order. An initial probability distribution L_P is devised based on the value of `CurrentFishPopulationSize`. For example, if `CurrentFishPopulationSize` > 40, then the initial probability distribution $L_P := [0.1, 0.3, 0.6]$. This implies $p(\text{low}) = 0.1, p(\text{medium}) = 0.3$, and $p(\text{high}) = 0.6$.

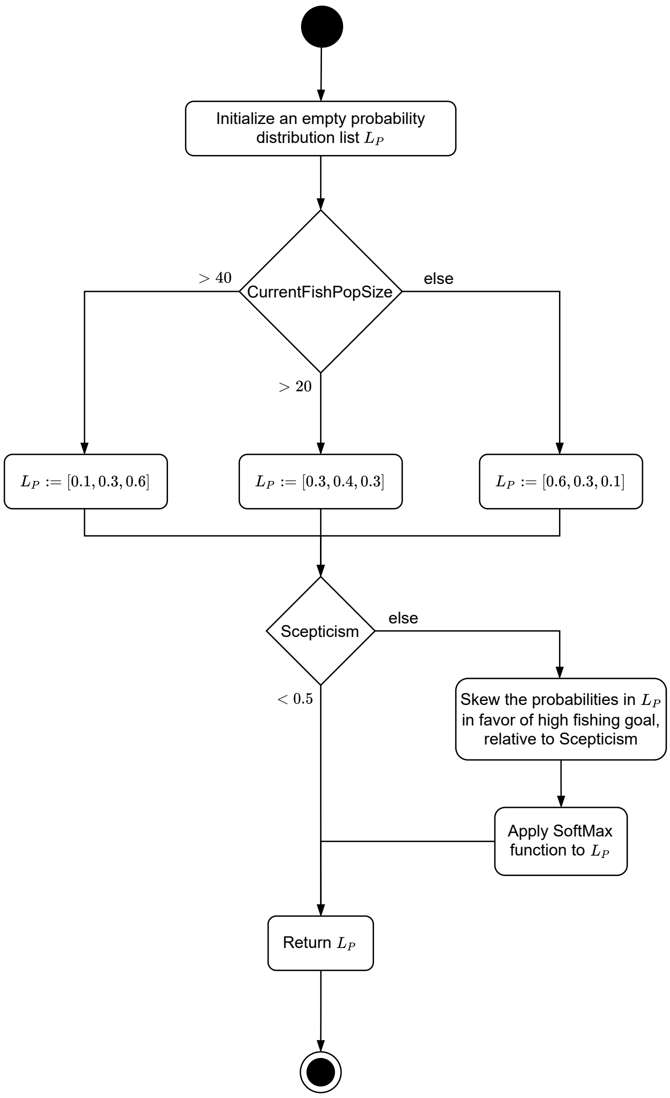

Figure 5.8: Activity diagram showing a Valuation processes performed in the `HuBccValuation` component (Example: `EvaluateFishingGoals()`)

These initial probabilities can be skewed by the agent's `Scepticism`. In the model, a probability distribution with three values is skewed by adding the value of `Scepticism` to each of its probabilities multiplying with 1, 2, and 4, respectively. The resulting numbers no longer add up to 1.0. To restore this property, the Softmax function σ is applied. σ is an order-preserving[6] function that normalises its inputs:

$$\sigma(L_P)_i = \frac{\mathrm{e}^{p_i}}{\sum_{j=1}^{|L_P|} \mathrm{e}^{p_j}}$$

For L_P as defined at the top of the previous page and `Scepticism` $= 0.6$, the probability distribution – after applying the process shown in Figure 5.8 – is $[0.1, 0.1, 0.8]$ (rounded to one significant digit). The new distribution more highly favours choosing a high fishing goal than L_P did.

The process of Choice is modelled by choosing an action from the set of available actions based on the probability that was computed for it by `HuBccValuation`. In the present example, $p(\text{low}) = 0.1, p(\text{medium}) = 0.1$, and $p(\text{high}) = 0.8$. A highly sceptical `Fisher` can still choose a moderate or low fishing goal under these circumstances, but the probability of doing so is small. In the case of evaluating the choosing fishing goals, furthermore, there is a small probability of choosing an extremely high goal, resulting in overfishing if the choice is executed.

Routine of the `FishPopulation`

Figure 5.9 shows the routine of the `FishPopulation`. At the end of each round, the agent updates the `CurrentPopulationSize` based on the applicable reproduction rate (see Section 4.3). Afterwards, and `CurrentReproductionRate` is updated according to the new population size. Furthermore, as stated in the rules of the CPR game, the population is restored to its full original size of 50 at the beginning of round 7.

[6] The i^{th} number in the natural order of the input before applying σ to L_P remains in the same place after applying σ. This is true for all numbers of the input.

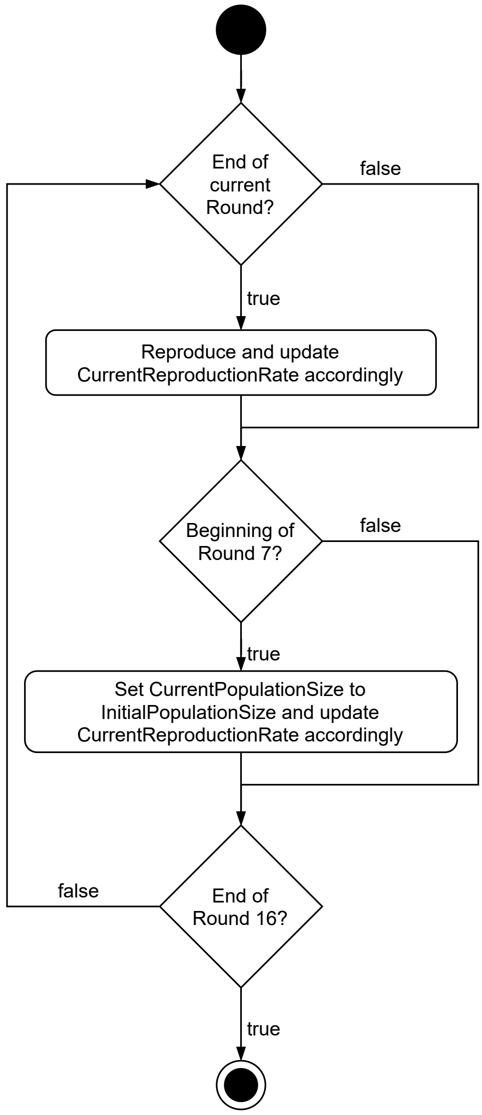

Figure 5.9: Activity diagram showing the behavioural routine of the `FishPopulation`

Routine of the **`Facilitator`**

As mentioned in Section 5.1, the `Facilitator` is responsible for clearing the collection `FishingGoals` in the layer type `Village`. This collection is used by `Fisher` agents to communicate their current fishing goals with each other. At the end of each round, the `Facilitator` empties the collection, making it ready for the next group communication. Figure 5.10 illustrates this routine.

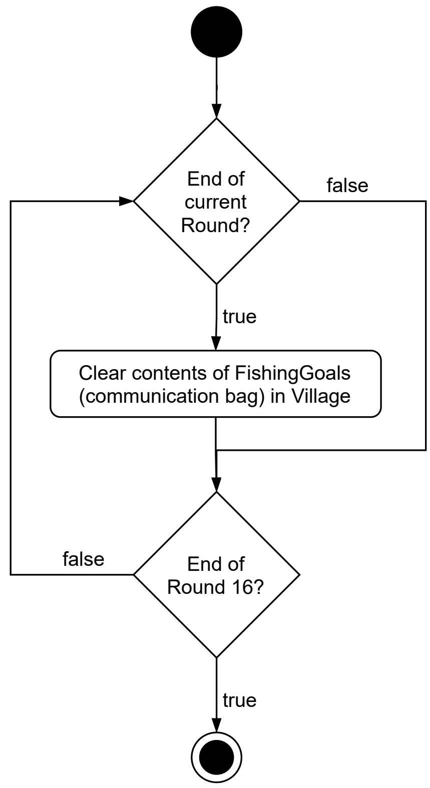

Figure 5.10: Activity diagram showing the behavioural routine of the `Facilitator`

5.5 Implementation Details

In this section, implementation details are listed that are considered to be relevant but did not fit in any of the preceding sections.

Refactoring

During the implementation process, it was noticed that some constants are used in several parts of the model. For example, the reproduction dynamics (see Figure 5.2) is accessible explicitly by the `FishPopulation` and implicitly by the `Fisher` via the `Mind`. Such multiple uses were identified and refactored into structs to serve as central points of access. This resulted in seven structs that hold different types of information. For instance, the probabilities used during the Valuation processes are stored in structs. This approach is thought to be less error-prone than storing values explicitly at their multiple locations of use, and can also be helpful during the modelling process. For instance, the values of a given probability distribution can be changed at one central location and automatically be applied at all locations of use throughout the model. For ABMs, it is possible that storing decision rules of agents in this fashion can also be worthwhile.

Model Configuration and Execution

The configuration of the model occurs via a JavaScript Object Notation (JSON) file in the project's root directory called config.json. The file contains configuration information that is typical for an ABM written in MARS[7] In addition, some model-specific parameters are required, such as the layer types `Village` and `FishingPlace`. Furthermore, the agent types `Fisher`, `FishPopulation`, and `Facilitator` with `count` values 4, 1, and 1, respectively, are required – the agent type `Weather` is optional. Each agent type's `outputFrequency` is set to 9, causing only one output to be produced per round.

In the directory Resources, the agent initialisation files are located. The file of the agent type `Fisher` is called init_fisher.csv. It contains values for the properties `Name`, `Trust`, `Reciprocity`, `Scepticism`, and `SocialNorm`. The initialisation file of the agent type `FishPopulation` is called init_fish_population.csv and contains initial values for the properties `InitialPopulationSize`, `CurrentPopulationSize`, and `ReproductionRate`. The agent type `Weather` has an initialisation file called init_-weather.csv that contains a value for the property `EventType`.

The simulation can be started via the file Program.cs that is located in the project's root directory. At the beginning of the `Main()` method, a local variable `numberOfSims` and `socialNorm` can be set to indicate the number of simulations that should be run and the social norm that should be active, respectively. **Note:** The value of `socialNorm` in the file Program.cs should be the same as the value of the property `SocialNorm` set in the initialisation file of the `Fisher` agent.

[7]For more details, please see `https://mars.haw-hamburg.de/articles/core/model-configuration/index.html`

6 Results

After designing and implementing the model as described in Chapter 5, it was executed using different configurations and initial values to study the `Fisher` agents' behaviour. In this chapter, the used configurations and values are first outlined before showing basic round-based, averaged data to verify the model's core functionality. This is followed by results that highlight overfishing and population extinction as well as the `Fisher` agents' development over time with respect to choosing positive trust-based and reciprocity-based actions. Lastly, the function for skewing probability distributions in favour of negative behaviours (see Section 5.4 and Figure 5.8) is reviewed.

6.1 Model Configuration

To produce the results shown in this chapter, the model was configured using the means described in Section 5.5. Table 6.1 shows the initial values for the agent type `Fisher` in the respective initialisation file. For other runs, the value of `SocialNorm` was changed to 1 and 2 for all four agents. A total of 100 runs were completed per `SocialNorm` value. Table 6.2 shows the initial values for the agent type `FishPopulation`.

Table 6.1: Initialisation parameters for the agent type `Fisher`

`Name`	`Trust`	`Reciprocity`	`Scepticism`	`SocialNorm`
A	1.0	1.0	0.0	0
B	1.0	1.0	0.0	0
C	1.0	1.0	0.0	0
D	1.0	1.0	0.0	0

Table 6.2: Initialisation parameters for the agent type `FishPopulation`

InitialFishPopulationSize	CurrentFishPopulationSize	CurrentReproductionRate
50	50	0

To evaluate simulation results, a Jupyter Notebook was prepared. It is in the project's repository in the directory Output. Simulation results that are produced by following the steps described in Section 5.5 can be retrieved and processed by running the entire notebook. All plots are produced using the Python library pyplot from matplotlib. In addition, the appendix that was submitted along with the thesis is available on CD and can be requested from the thesis' first supervisor.

6.2 Model Validation

The following three plots show the `FishCount` per agent per round, averaged over 100 simulation runs. In addition, the average `FishPopulationSize` is shown. These plots aim to serve for basic model validation by addressing whether the model adequately performs the most basic functionality it is intended for, i.e., producing a plausible dynamic between increasing `FishCounts` and a decreasing `FishPopulationSize`.

Figure 6.1 shows the development of these properties when `SocialNorm` = 0 (see Section 5.2). In this setting, agents to not engage in any communication and instead fish by themselves.

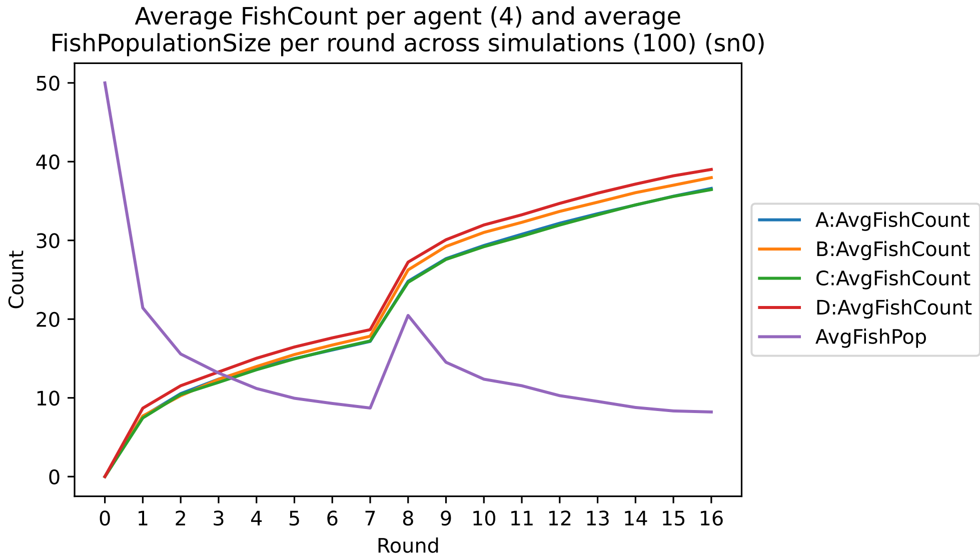

Figure 6.1: FishCount and FishPopulationSize per round, averaged over 100 simulations (SocialNorm = 0)

As defined in the initialisation file, the FishPopulationSize starts at 50, while the Fisher agents start the simulation with 0 fish. The population size declines steeply. The FishCount values experience an initially steep increase, while begins to taper off after 1-2 rounds. The system seems to stabilise until the FishPopulationSize is restored to 50 at the beginning of round 7. (The restoration is not seen in the plot because the FishPopulationSize is decreased again by the Fisher agents before its value is written to the output file). Between rounds 7-9, a similar trend to the one seen between rounds 0-2 can be observed. For the remainder of the game, the Fisher agents seem to fish conservatively, allowing the FishPopulationSize to stabilise. The fishing behaviour of the Fisher agents appears to be very similar to each other. It should be restated that these results were obtained by averaging values from over 100 simulation, potentially averaging out outlier behaviours.

Figure 6.2 and Figure 6.3 show results for the same experimental setup, but with SocialNorm = 1 and SocialNorm = 2, respectively. The trends are similar to the one seen for SocialNorm = 0. For SocialNorm = 1, a steeper average decline of the FishPopulationSize can be observed near the end of the simulation. For SocialNorm = 2, the FishCount values appear even more closely aligned than when SocialNorm = 0 or SocialNorm = 1.

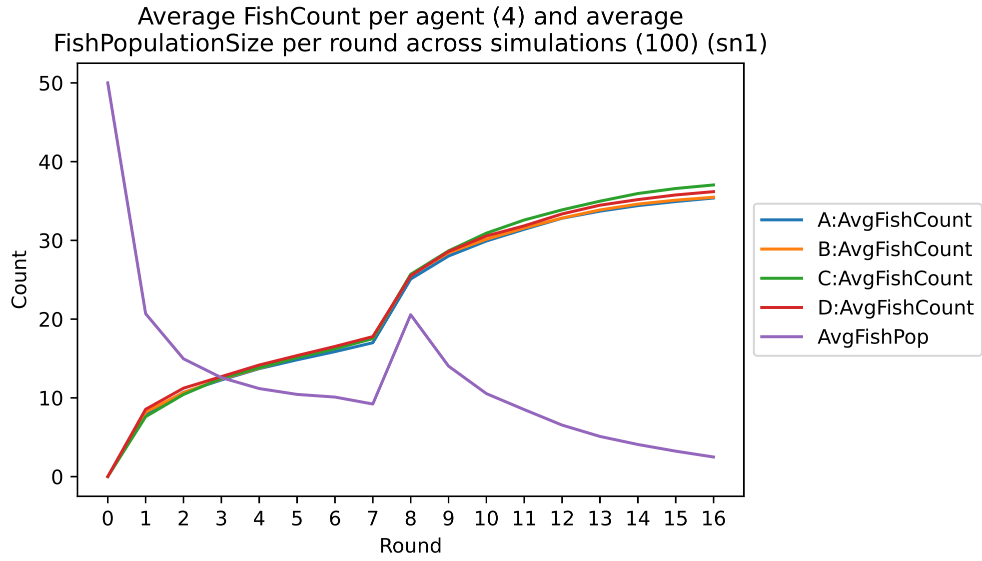

Figure 6.2: `FishCount` and `FishPopulationSize` per round, averaged over 100 simulations (`SocialNorm` $= 1$)

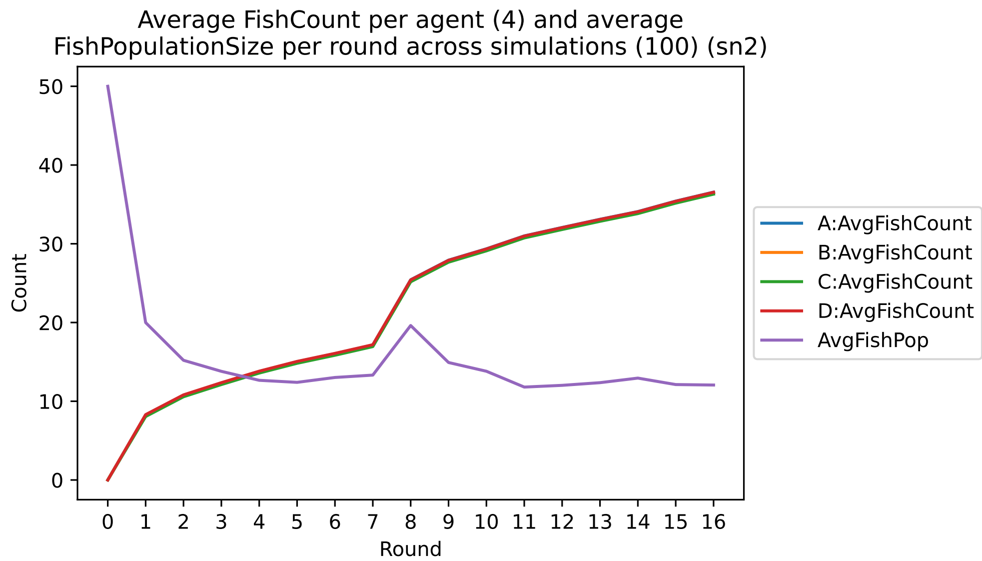

Figure 6.3: `FishCount` and `FishPopulationSize` per round, averaged over 100 simulations (`SocialNorm` $= 2$)

6.3 Overfishing and Extinction

A important theme of CPR games is the potential extinction of the natural resource in question. Figure 6.4 shows the absolute counts of overfishing events (see Section 5.4) and extinction events across 100 simulations, stratified by the active `SocialNorm`.

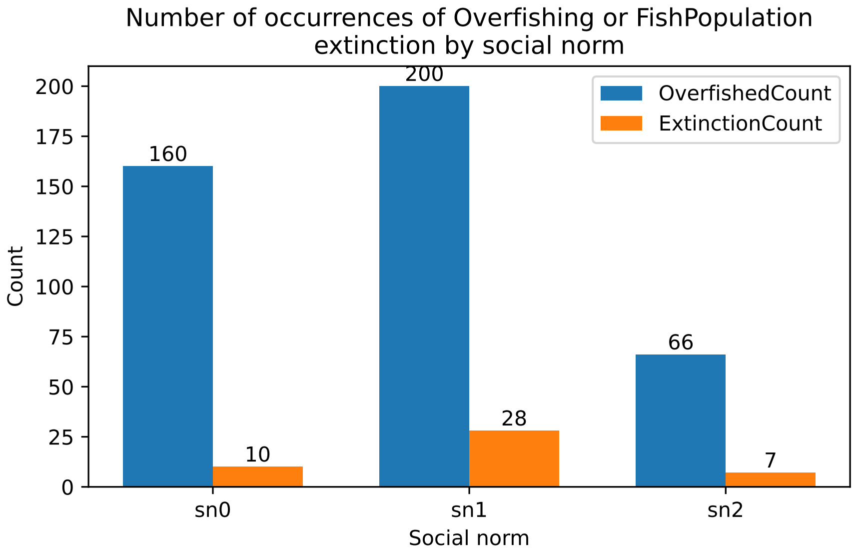

Figure 6.4: Number of times overfishing or population extinction occurred, averaged over 100 runs per `SocialNorm`.

In the individualistic setting (`SocialNorm` = 0), there were 160 overfishing events and 10 extinction events, while in the initially collectivistic setting (`SocialNorm` = 1), there were 200 and 28, respectively. Lastly, in the fully collectivistic setting (`SocialNorm` = 2), there were 66 overfishing events and 7 extinction events. While there seem to be clear differences in behaviour that cause these numbers, it is important to note that, altogether, the agents had $4 \times 16 \times 100 = 6{,}400$ opportunities to engage in behaviour that results in overfishing and/or population extinction. Table 6.3 shows the corresponding relative frequencies.

Table 6.3: Relative frequencies of overfishing and population extinction during 100 simulation runs, stratified by active social norm

Metric	`SocialNorm` = 0	`SocialNorm` = 1	`SocialNorm` = 2
Overfishing	0.025	0.03	0.01
Extinction	0.002	0.004	0.001

6.4 Trust, Reciprocity, Scepticism

The `Fisher` agents' internal logic is strongly focused on modelling valuation and choice processes. Decision-making is a key element of both the model and the original CPR game. Figure 6.5 shows the ratio between, on the one hand, positive trust-based/reciprocity-based behaviours (i.e., communicating one's fishing goal to the community and adhering to the communicated fishing goal, respectively) and the total number of such decisions. On the other hand, it shows the ratio between negative trust-based/reciprocity-based behaviours (i.e., not communicating and not adhering to the communicated fishing goal, respectively) and the total number of such decisions. These data are available only for simulations with `SocialNorm` = 1, since the distinction between positive and negative trust-based and reciprocity-based behaviours do not apply in the other two variations.

While the chart features many line plots and is difficult to read in detail, the main trend can be stated as follows: the `Fisher` agents began the simulation with high trust (top left of the plot area) because their initial values were configured that way (see Table 6.1). However, during the simulation, their willingness to trust and/or reciprocate decreases as a result of increasing scepticism. Figure 6.6 shows a corresponding linear increase of `Scepticism` over time (also for `SocialNorm` = 1).

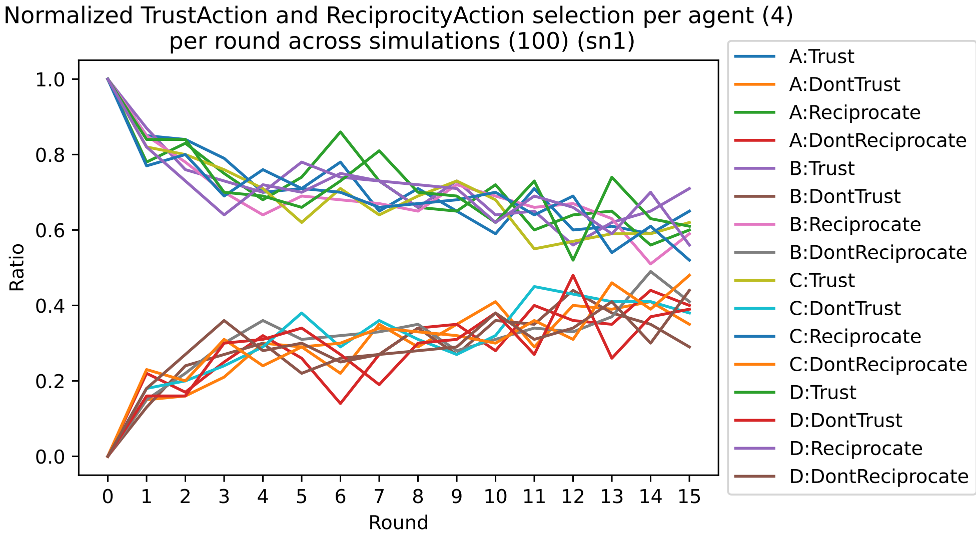

Figure 6.5: Ratio between positive/negative trust-/reciprocity-based behaviours and total number of behaviours per agent over rounds, averaged over 100 simulation runs (SocialNorm = 1)

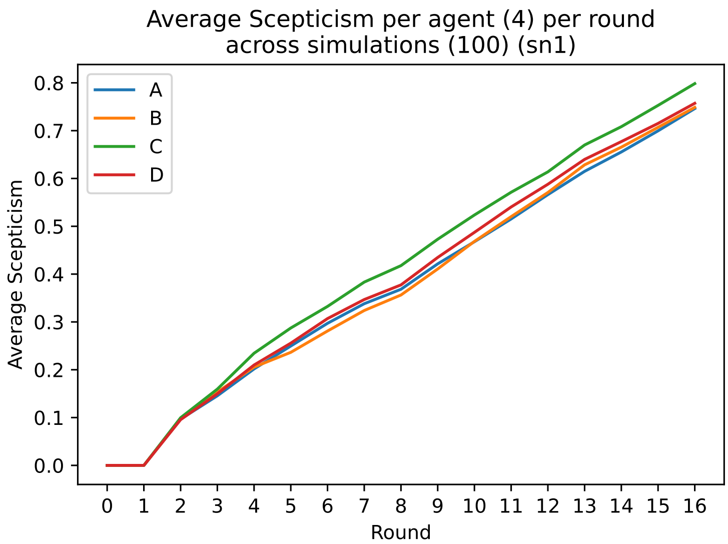

Figure 6.6: Scepticism trend per agent over rounds, averaged over 100 runs (SocialNorm = 1)

6.5 Skewing Probability Distributions

The following four plots show the trajectory of the skewing function that was introduced in Section 5.4. Based on an an agent's current value of `Trust/Reciprocity` and `Scepticism`, the function skews a probability distribution over a set of choices towards the negative choice(s). As mentioned in Section 5.2, both `Trust` and `Reciprocity` are modelled as stable characteristics, while `Scepticism` is modelled as a situational characteristic. Intuitively, it is expected that the lower any of these values, the more strongly the function skews a given probability distribution towards negative choices (i.e., not trusting or not reciprocating).

The plots show the probability of choosing an action as a function of `Scepticism`, with `Scepticism` ranging from 0.0 to 1.0. Figure 6.7 shows the trajectory of the function when `Trust/Reciprocity` is fixed at 0.25. Focusing on the blue line (the positive behaviour), it can be seen that the probability of choosing it increases slightly as `Scepticism` increases from 0.0 to 0.1, before proceeding to decrease as `Scepticism` approaches 1.0. The small increase at the beginning is an inaccuracy in the function.

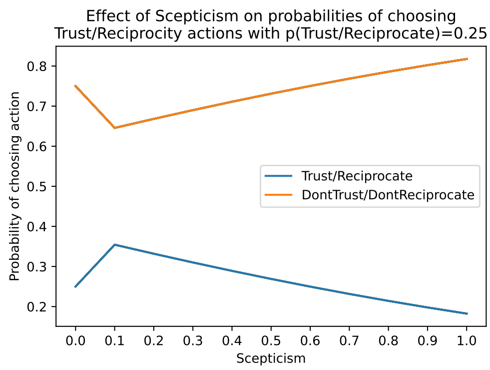

Figure 6.7: Trajectory of skewing function with `Trust/Reciprocity` $= 0.25$

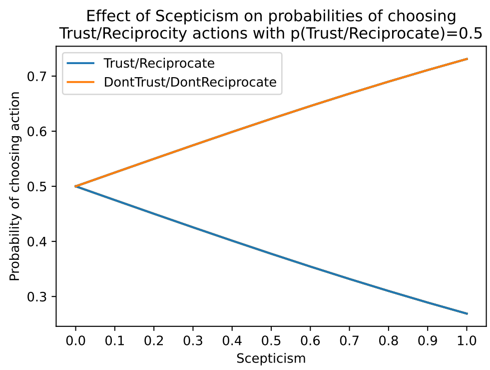

Figure 6.8: Trajectory of skewing function with `Trust/Reciprocity` $= 0.5$

Figure 6.8 shows the trajectory that emerges when `Trust/Reciprocity` is fixed at 0.5. When `Scepticism` $= 0.0$, the probabilities of choosing the positive or negative behaviour are equal. As `Scepticism` increases, the probability of choosing the positive action decreases. Figure 6.9 and Figure 6.10 show the trajectory for `Trust/Reciprocity` $= 0.75$ and `Trust/Reciprocity` $= 1.0$, respectively.

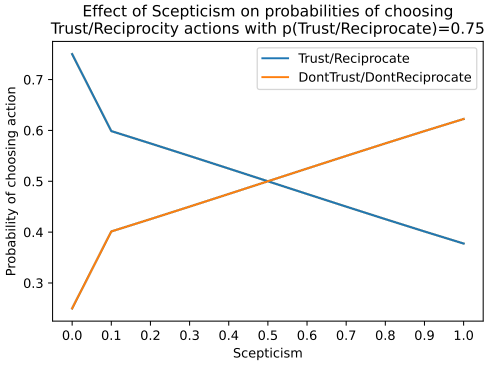

Figure 6.9: Trajectory of skewing function with `Trust`/`Reciprocity` $= 0.75$

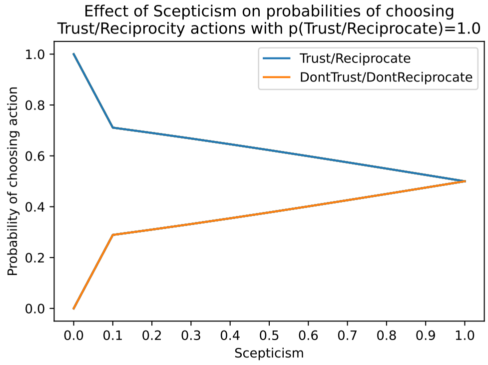

Figure 6.10: Trajectory of skewing function with `Trust`/`Reciprocity` $= 1.0$

7 Discussion and Conclusion

In this final chapter of the thesis, an evaluation of the results is offered. This is followed by a review of the internal structure of the agent in conjunction with the application of the HuB-CC framework's elements. The interdisciplinary nature of implementing theory-based behavioural routines in agents is touched on, before concluding with an outlook to potential future endeavours that might build on the findings and learnings of this thesis.

7.1 Analysis of Results

This section offers a critical analysis of the results presented in Chapter 6. In Figure 6.1, Figure 6.2, and Figure 6.3, the trajectories of the agents' `FishCount` values are very consistent, both within one `SocialNorm` and across `SocialNorms`. This seems to indicate that the agents' decision-making and behaviour is unaffected by the active `SocialNorm`. However, the plots of these three figures are based on average values obtained from 100 simulation runs per `SocialNorms`. It appears that, while there is variance across individual decisions and behaviours, it is negligible enough to be averaged out over a large number of simulation runs. Hence, while the three figures illustrate the general course of a simulation (in terms of `FishCount` and `FishPopulationSize`), they do not illustrate outlier behaviours. Instead, they serve to verify the model – as they show that `Fisher` agents, on average, behave as they should and the `FishPopulation` decreases as it is expected to – and show the model's trend over time. It is worth noting, however, that for `SocialNorm` = 0, the four trajectories differ most, albeit still not significantly.

Figure 6.4 more aptly shows outlier behaviours. While the rate of overfishing and population extinction is small, these behaviours are exhibited occasionally by the agents. Table 6.3 shows that the rates are extremely small when considering the total number of decisions made by the agents that could potentially result in overfishing or population extinction over 100 simulation runs. In summary, the model is descriptive enough to produce these behaviours. But it seems that the details of decision-making need to be fine-tuned in order to produce more realistic rates. This would have been possible if there were similar numerical measurements shown from the CPR game in [31] for the simulation to aim for.

A similar assessment can be made about the trajectories of the ratios of `Trust`- and `Reciprocity`-based actions shown in Figure 6.5 as well as the value of `Scepticism` shown in Figure 6.6. The latter shows an almost linear increase, indicating that the model, on average, simply increments `Scepticism` by an almost constant value. Intuitively, a more realistic growth of `Scepticism` might take the shape of logarithmic function that converges to some upper bound that is less than 1.0. At the rate seen in Figure 6.6, the value of `Scepticism` would likely reach 1.0 and remain there if the simulation continued for more rounds. It is doubtful that human scepticism behaves this way, and it cannot be inferred from the theory of Trust and Reciprocity because it does not give insights into long-term interactions between individuals.

Intuitively, the trajectories of the skewing functions shown in Figure 6.7 through Figure 6.10 appear plausible. The higher the value of `Trust`/`Reciprocity` (see, for example, Figure 6.9 and 6.10), the higher the starting point of the blue line (which indicates a choice of a trusting/reciprocating action). As `Scepticism` increases, the probability of making that choice decreases, while the probability of making the opposite (negative) choice increases proportionally. This function appears to describe the relationship between the three variables well – as it is defined in the model, which is not fully supported by the underlying theories of human behaviour. The only area of concern in the trajectory is around `Scepticism` = 0.1, where there is a kink in three of the four lines. This can potentially be smoothed out by tweaking the function further.

Finally, it must be noted that more simulation runs with different configuration values could have been performed to showcase a more varied scope of behaviour that the model can potentially produce. For example, the agents could have been initialised with lower values for `Trust`/`Reciprocity` to observe the effect on the course of the simulation. As shown in Table 6.1, such runs were not completed for this thesis.

7.2 Analysis of Agent Structure and HuB-CC elements

As outlined in detail in Section 5.3, an attempt was made to integrate the elements of the HuB-CC framework into the internal structure of the `Fisher` agent. A literal representation of the HuB-CC elements was chosen for this purpose, meaning that each element that was mapped to by one of the theories (see Section 4.4) was included as one class in the architecture. To facilitate easy communication between the `Fisher` agent and its `Mind`, the façade pattern was implemented. As a result, the `Mind` provides a uniform interface via a set of public methods to the `Fisher` agent. This reduces the number of dependencies in the `Fisher`-`Mind` relationship and eliminates any need for the `Fisher` to be aware of the internal structure of the `Mind`.

Throughout the design and implementation process, the elements of the HuB-CC framework proved very useful as an organisational tool. From a modeller's perspective, it is helpful to think of an agent's activities in terms of the elements to facilitate a clear separation of concerns in the design. This is also useful for assessing the sophistication and completeness of an agent's behavioural routine with respect to specific elements. If, for example, it is found that the agent's Valuation component is insufficient in some way, it is easy to identify the place in the model at which adjustments need to be made because of the clear separation facilitated by the elements. While a one-to-one representation of the HuB-CC elements was implemented in this model, it can be argued that certain elements can be grouped together to simplify the architecture. For example, the elements Stable Characteristics and Situational Characteristics can be grouped into one class and separated internally via some syntactic means.[1] Likewise, it is possible that combining the elements Perception and Social and Biophysical Environment can simplify certain routines that the agent – or, in case of architecture designed for this thesis, the `Mind` – needs to execute repeatedly. Perception, by design, is likely to often occur in conjunction with aspects of the Social and Biophysical Environment. Such simplifications can improve the code base while still maintaining the domain-specific separation of concerns provided by the HuB-CC framework.

The HuB-CC framework as a whole was found to be helpful in informing the design process. Without the use of theories, much of the design of agent behaviour is left to the modeller's intuition and best guess. The theories help the modeller to ground her design

[1] In C#, for example, such a separation can be achieved by dividing the code base into regions.

decision in a theoretical foundation. Still, it remains challenging to choose a well-suited theory and to abstract the necessary information from it to devise a strong design.

7.3 Interdisciplinary Work

One of the intentions and requirements (see Chapter 3) of this work was to develop an ABM through interdisciplinary work with domain experts from the social/cognitive sciences and sustainability science. This goal was achieved in so far as the conceptual portion at the beginning of the thesis, i.e., before the design and implementation process, was carried out over a number of conversations with psychologists and a sustainability expert. The model basis as well as the theories were identified during this stage. The remainder of the project was carried out by the author of this thesis.

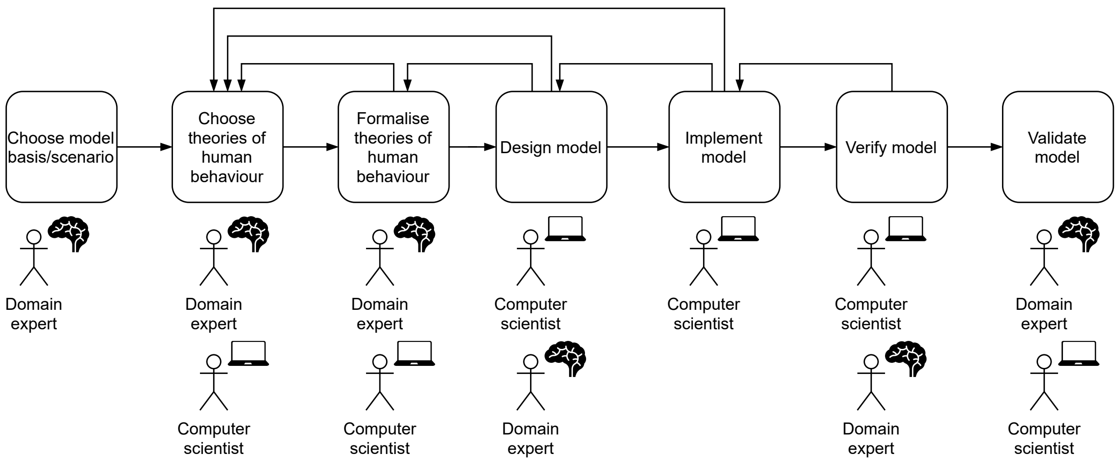

Figure 7.1: Proposed interdisciplinary workflow between domain experts and computer scientists to design a theory-driven ABM

Figure 7.1 shows a proposed interdisciplinary workflow in which an ABM with theory-oriented behavioural logic can be implemented from the initial conception to the finished model. In the diagram[2], the high-level stages of the modelling process that were carried out for this thesis are listed in sequential order. Below each stage, either one or two

[2]Syntactically, this diagram uses a loose adaptation of the Unified Modelling Language (UML) activity diagram. It is not intended, nor does it claim, to use that syntax accurately.

actors are present, indicating the role of the individual[3] who is intended to perform the task. When two actors are present, the actor that is closer to the task is considered to be the main performer of the task, while the other actor serves at least a supportive function. The main path is linear from left to right; however, it is important to note the feedback loops, which indicate that the process might involves steps back to passed stages to reconsider previously made selections (e.g., of theories) and design decisions (e.g., within the model).

For this thesis, the first two steps shown in Figure 7.3 were carried out with the help of domain experts. As the diagram shows, a longer-term interaction between the computer scientist and the domain expert who participate in the project is thought to be favourable to ensure a satisfactory end result. At the end of the workflow, the two stages "Verify model" and "Validate model" indicate the confirmation of the functional correctness of the model and the non-functional correctness of the model, respectively. In other words, the computer scientist is mainly responsible for confirming that the model performs correctly (i.e., does not contain any computational errors or other problems that were introduced during the implementation phase). The domain expert, on the other hand, is better suited for confirming that the model performs accurately in terms of the theory formalisations that were defined at the beginning of the process.

7.4 Conclusion and Outlook

This project involved a theory-oriented design and implementation of agent behaviour in an ABM, given a model scenario that is based on a real-world CPR. While there were many valuable lessons and takeaways for the author of this thesis, the integration of all of these parts proved challenging. From a non-domain expert's perspective, some aspects of the chosen theories were found to be not ideally suited for the model scenario. Likewise, the theories did not touch on some aspects that were thought to be necessary for fully describing the agents' behaviour in a intended theory-oriented fashion. As a result, the theory formalisations (see Section 5.2) are rudimentary. It is apparent that a strong and well-matched pairing between a model scenario and a set of theories is crucial for a fruitful implementation.

[3] For simplicity, the written description uses the singular form. But each actor in Figure 7.1 can represent either one person or a group of persons, depending on whether a single computer scientist/domain expert or a group is involved in the project.

Still, the usefulness of consulting theories of human behaviour when implementing behavioural routines of agents that are meant to represent humans is clear. Also, the segmentation of an agent's internal structure along domain-informed lines helps to facilitate a clear and clean internal architecture. While the architecture devised in this work is in some respects not ideal (as touched on in Section 5.3 and Section 7.2), it can be improved on in future iterations.

Bibliography

[1] Abar, S., Theodoropoulos, G. K., Lemarinier, P., and O'Hare, G. M. Agent based modelling and simulation tools: A review of the state-of-art software. *Computer Science Review 24* (2017), 13–33.

[2] Altmann, S., Dohmen, T., and Wibral, M. Do the reciprocal trust less? *Economics Letters 99*, 3 (2008), 454–457.

[3] Anderson, E., and Mammides, C. The role of protected areas in mitigating human impact in the world's last wilderness areas. *Ambio 49*, 2 (2020), 434—441.

[4] Berg, J., Dickhaut, J., and McCabe, K. Trust, reciprocity, and social history. *Games and Economic Behavior 10*, 1 (1995), 122—142.

[5] Bicchieri, C. Measuring social norms. https://irh.org/wp-content/uploads/2016/09/Bicchieri_MeasuringSocialNorms.pdf.

[6] Bicchieri, C. *The Grammar of Society: The Nature and Dynamics of Social Norms*. Cambridge University Press, 2006.

[7] Bicchieri, C. Why people do what they do?: A social norms manual for zimbabwe and swaziland. Innocenti Toolkit Guide from the UNICEF Office of Research, Oct. 2015.

[8] Bicchieri, C. *Norms in the Wild: How to Diagnose, Measure, and Change Social Norms*. Oxford University Press, 2016.

[9] Clemen, T., Ahmady-Moghaddam, N., Lenfers, U. A., Ocker, F., Osterholz, D., Ströbele, J., and Glake, D. Multi-agent systems and digital twins for smarter cities. In *Proceedings of the 2021 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation* (2021), vol. 1, ACM, pp. 45—-55.

[10] Constantino, S., Schlüter, M., Weber, E., and Wijermans, N. Cognition and behavior in context: a framework and theories to explain natural resource use decisions in social-ecological systems. *Sustainability Science 16* (Sep. 2021), 1651—1671.

[11] Cuevas, E. An agent-based model to evaluate the covid-19 transmission risks in facilities. *Computers in Biology and Medicine 121* (2020), 103827.

[12] Dawson, R. J., Peppe, R., and Wang, M. An agent-based model for risk-based flood incident management. *Natural Hazards 59*, 1 (2011), 167–189.

[13] de Waard, D., Toffetti, A., Pietrantoni, L., Franke, T., Petiot, J.-F., Dumas, C., Botzer, A., Onnasch, L., Milleville, I., and Mars, F., Eds. *Understanding Human Behaviour in Complex Systems: Proceedings of the Human Factors and Ergonomics Society Europe Chapter 2019 Annual Conference* (May 2020), HFES, HFES. HFES Europe Chapter conference: Understanding Human Behaviour in Complex Systems; Conference date: 02-10-2019 through 04-10-2019.

[14] Dobbie, S. L., Schreckenberg, K., Dyke, J., Schaafsma, M., and Balbi, S. Agent-based modelling to assess community food security and sustainable livelihoods. *Journal of Artificial Societies and Social Simulation 21(1)*, 1 (January 2018), 1–23.

[15] Dohmen, T., Falk, A., Huffman, D., and Sunde, U. Representative trust and reciprocity: Prevalence and determinants. *Economic Inquiry 46*, 1 (2008), 84–90.

[16] Dressler, G., Groeneveld, J., Buchmann, C. M., Guo, C., Hase, N., Thober, J., Frank, K., and Müller, B. Implications of behavioral change for the resilience of pastoral systems—–lessons from an agent-based model. *Ecological Complexity 40* (2019), 100710. Agent-based modelling to study resilience in socio-ecological systems.

[17] Fagiolo, G., Windrum, P., and Moneta, A. Empirical validation of agent-based models: A critical survey. LEM Working Paper Series 2006/14, Laboratory of Economics and Management (LEM), Pisa, 2006.

[18] Gamma, E., Helm, R., Johnson, R. E., and Vlissides, J. *Design Patterns. Elements of Reusable Object-Oriented Software.* Prentice Hall, Indianapolis, IN, 1997.

[19] Glake, D., Ritter, N., Ocker, F., Ahmady-Moghaddam, N., Osterholz, D., Lenfers, U., and Clemen, T. *Hierarchical Semantics Matching For Heterogeneous Spatio-Temporal Sources.* Association for Computing Machinery, New York, NY, USA, 2021, p. 565–575.

[20] Goudie, A. S. *Human Impact on the Natural Environment*, 8 ed. Wiley-Blackwell, Oxford, UK, 2018.

[21] Heath, B., Hill, R., and Ciarallo, F. A survey of agent-based modeling practices (january 1998 to july 2008). *Journal of Artificial Societies and Social Simulation 12*, 4 (2009), 9.

[22] Holden, R. J., Scott, A. M. M., Hoonakker, P. L. T., Hundt, A. S., and Carayon, P. Data collection challenges in community settings: insights from two field studies of patients with chronic disease. *Quality of Life Research 24* (May 2015), 1043—1055.

[23] Imkamp, T., and Pross, S. *Differentialgleichungen für Einsteiger – Grundlagen und Anwendungen mit vielen Übungen, Lösungen und Videos*, 1. auflage ed. Springer-Verlag GmbH, 2019.

[24] Janssen, M. A. Impact of diverse behavioral theories on environmental management: Explorations with daisyworld. In *2016 Winter Simulation Conference (WSC)* (Dec. 2016), Institute of Electrical and Electronics Engineers (IEEE), Institute of Electrical and Electronics Engineers (IEEE), pp. 1690—1701.

[25] Janssen, M. A., and Baggio, J. A. Using agent-based models to compare behavioral theories on experimental data: Application for irrigation games. *Journal of Environmental Psychology 52* (June 2017), 194—203.

[26] Lenfers, U. A., Ahmady-Moghaddam, N., Glake, D., Ocker, F., Osterholz, D., Ströbele, J., and Clemen, T. Improving model predictions—integration of real-time sensor data into a running simulation of an agent-based model. *Sustainability 13*, 13 (2021).

[27] Lenfers, U. A., Ahmady-Moghaddam, N., Glake, D., Ocker, F., Ströbele, J., and Clemen, T. Incorporating multi-modal travel planning into an agent-based model: A case study at the train station kellinghusenstraße in hamburg. *Land 2021 10*, 11 (2021).

[28] Monroe, A., Olapeju, B., Moore, S., Hunter, G., Merritt, A. P., Okumu, F., and Babalola, S. Improving malaria control by understanding human behaviour. *Bulletin of the World Health Organization 99* (Nov. 2021), 837—839.

[29] Muelder, H., and Filatova, T. One theory—many formalizations: Testing different code implementations of the theory of planned behaviour in energy agent-based models. *Journal of Artificial Societies and Social Simulation 4*, 21 (Oct. 2018).

[30] Nikolai, C., and Madey, G. Tools of the trade: A survey of various agent based modeling platforms. *Journal of Artificial Societies and Social Simulation 12*, 2 (2009), 2.

[31] Rocha, J. C., Schill, C., Saavedra-Díaz, L. M., Moreno, o. d. P., and Maldonado, J. H. Cooperation in the face of thresholds, risk, and uncertainty: Experimental evidence in fisher communities from colombia. *PLOS ONE 15*, 12 (Dec. 2021), 1—14.

[32] Schill, C., Lindahl, J. M. A. T., Folke, C., Polasky, S., Cárdenas, J. C., Crépin, A.-S., Janssen, M. A., Norberg, J., and Schlüter, M. A more dynamic understanding of human behaviour for the anthropocene. *Nature Sustainability 2* (Dec. 2019), 1075—1082.

[33] Schlüter, M., Baeza, A., Dressler, G., Frank, K., Groeneveld, J., Jager, W., Janssen, M. A., McAllister, R. R., Müller, B., Orach, K., Schwarz, N., and Wijermans, N. A framework for mapping and comparing behavioural theories in models of social-ecological systems. *Ecological Economics 131* (2017), 21—35.

[34] Schwarz, N., Dressler, G., Frank, K., Jager, W., Janssen, M., Müller, B., Schlüter, M., Wijermans, N., and Groeneveld, J. Formalising theories of human decision-making for agent-based modelling of social-ecological systems: practical lessons learned and ways forward. *Socio-Environmental Systems Modelling 2* (Dec. 2020), 16340.

[35] Simon, H. A. A behavioral model of rational choice. *The Quarterly Journal of Economics 69*, 1 (1955), 99—118.

[36] Simon, H. A. *Models of man: social and rational.* Wiley, Oxford, 1957.

[37] Simon, H. A. Theories of bounded rationality. In *Decisions and Organizations*, C. McGuire and R. Radner, Eds. North-Holland Publishing Company, 1972, ch. 8, pp. 161—176.

[38] Sutton, R. S., and Barto, A. G. *Reinforcement Learning: An Introduction*, 2^{nd} ed. ed. The MIT Press, 2018.

[39] Tyldum, G., and Brunovskis, A. Describing the unobserved: Methodological challenges in empirical studies on human trafficking. *International Migration 43*, 1–2 (2005), 17–34.

[40] von Neumann, J., and Morgenstern, O. *Theory of Games and Economic Behavior: 60th Anniversary Commemorative Edition.* Golden Keys Success, 2020.

[41] Weyl, J., Lenfers, U., Clemen, T., Glake, D., Panse, F., and Ritter, N. Large-scale traffic simulation for smart city planning with mars. In *Proceedings of the 2019 Summer Simulation Conference* (2020).

[42] Wijermans, N., Boonstra, W. J., Orach, K., Hentati-Sundberg, J., and Schlüter, M. Behavioural diversity in fishing—towards a next generation of fishery models. *Fish and Fisheries 21*, 5 (2020), 872—890.

Glossary

Attention A HuB-CC element that describes the process focused on perception, information acquisition and updating, and learning [10].

Behaviour A HuB-CC element that describes the process by which an individual executes the choice made [10].

Bounded Rationality A theory of human behaviour that describes the human decision-making process as being rational (i.e., focused on maximising personal utility) and bounded (i.e., limited in knowledge and information-processing capability by conditions and constraints imposed by the environment) (see [36]).

Choice A HuB-CC element that describes the process by which an individual chooses an option or action [10].

External Information Search A HuB-CC element that describes the process of acquiring information through an external search [10].

Learning and Updating A HuB-CC element that describes the process by which an individual updates her mental representations and characteristics and learns about rules, norms, or novel events and environments [10].

Memory Search A HuB-CC element that describes the process of recalling, retrieving, and recognising information that has been previously encoded [10].

Perception A HuB-CC element that describes the process by which an individual senses and interprets the surrounding social and biophysical environment [10].

Rational Choice A theory of human behaviour that describes the human decision-making process as being rational, i.e., focused on maximising personal utility.

Satisficing The act of foregoing a choice that might yield high short-term reward to enable an even higher long-term reward, usually in the face of incomplete information.

Situational Characteristics A HuB-CC element that describes characteristics that are activated or salient at a given moment in time or in a specific context (typically subsets of Stable Characteristics) [10].

Social and Biophysical Environment A HuB-CC element that describes the environment in which an individual is situated [10].

Social Norms A theory of human behaviour that describes a connection between norms at the societal level and expectations at the individual level, aiming to identify social norms in communities and moderate expectations of members of that community to facilitate a desired change in their behaviour (see [6, 8]).

Stable Characteristics A HuB-CC element that describes an individual's personality traits, dispositions, and other variables that can be considered permanent, chronically present, or slowly changing [10].

Trust and Reciprocity A theory of human behaviour that describes how trust and reciprocity factor into a one-time, two-way exchange between two individuals that do not know each other (i.e., interact with each other for the first time) (see [4]).

Valuation A HuB-CC element that describes the process by which an individual evaluates the qualities or desirability of an option [10].

Declaration of Independent Work on a Thesis

I hereby certify that I have written this thesis independently without outside help and have used only the indicated aids. Passages taken verbatim or in spirit from other works are indicated with reference to the sources.

Erklärung zur selbstständigen Bearbeitung einer Abschlussarbeit

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

Hamburg, Germany	14 January 2022	
Place/Ort	Date/Datum	Original Signature/Unterschrift im Original