A Software Architecture for Mechanism-Based Social Systems Modelling in Agent-Based Simulation Models

Tuong Manh Vu[1]**, Charlotte Probst**[2]**, Alexandra Nielsen**[3]**, Hao Bai**[4]**, Charlotte Buckley**[4]**, Petra S. Meier**[1]**, Mark Strong**[1]**, Alan Brennan**[1]**, Robin C. Purshouse**[4]

[1]*School of Health and Related Research, University of Sheffield, Regent Court 30 Regent Street, Sheffield S1 4DA, United Kingdom*
[2]*Heidelberg Institute of Global Health, Universitätsklinikum Heidelberg, Im Neuenheimer Feld 130.3, Heidelberg 69120, Germany*
[3]*Alcohol Research Group, 6001 Shellmound St. Ste 450, Emeryville CA 94608, United States*
[4]*Department of Automatic Control and Systems Engineering, University of Sheffield, Mappin Street, Sheffield S1 3JD, United Kingdom*
Correspondence should be addressed to t.vu@sheffield.ac.uk

Abstract: This paper introduces the MBSSM (Mechanism-Based Social Systems Modelling) software architecture that is designed for expressing mechanisms of social theories with individual behaviour components in a unified way and implementing these mechanisms in an agent-based simulation model. The MBSSM architecture is based on a middle-range theory approach most recently expounded by analytical sociology and is designed in the object-oriented programming paradigm with Unified Modelling Language diagrams. This paper presents two worked examples of using the architecture for modelling individual behaviour mechanisms that give rise to the dynamics of population-level alcohol use: a single-theory model of norm theory and a multi-theory model that combines norm theory with role theory. The MBSSM architecture provides a computational environment within which theories based on social mechanisms can be represented, compared, and integrated. The architecture plays a fundamental enabling role within a wider simulation model-based framework of abductive reasoning in which families of theories are tested for their ability to explain concrete social phenomena.

Keywords: Agent-Based Modelling, Social Simulation, Software Architecture, Analytical Sociology, Abductive Reasoning

Introduction

Mechanism-based social systems modelling

1.1 Social system models are designed to study and aid our understanding of the social world. In the analytical sociology tradition, social system models explain social phenomena by explicitly defining the mechanisms of action and interaction that bring them about (Hedström & Bearman 2009). Elster elucidates: "A mechanism explains by opening up the black box and showing the cogs and wheels of the internal machinery. A mechanism provides a continuous and contiguous chain of causal or intentional links between the explanans and the explanandum" (Elster 1989). Thus analytical social system models seek to explain *why* things happen through a series of causal mechanisms: a mechanism-based *theory*.

1.2 Social theorists have developed a number of meta-models for mechanism-based theories. What typically differentiates these meta-models is the epistemological position that underpins their formulations, but they do

share common general features. The meta-models are comprised of levels: the micro-level (which may be comprised of individuals or, more generally, entities which exhibit agency) and the macro level (which may be social structures, social emergents, or social level entities such as institutions). Mechanisms operate across or within levels: social entities affect micro entities (macro-micro), an individual micro entity affects another micro entity (micro-micro), micro entities' internal states result in action (micro-micro), or actions of micro entities affect social entities (micro-macro), and social entities affect other social entities (macro-macro). We will refer to this general meta-model as the "general micro-macro scheme." Below we review the most significant examples of such schemes –– we focus first on the approach most commonly associated with the paradigm of analytical sociology (Hedström & Bearman 2009), then we introduce the alternative proposals.

The Coleman boat

1.3   Coleman (1986) developed a meta-model of mechanisms linking macro, social-level entities or phenomena to micro, agency-level actions, that is colloquially known as the "Coleman boat" (Ylikoski 2016). The "agency level" is often taken to mean individuals or persons, but this should not be assumed. In Coleman's models, only micro-level entities, whether they are individuals of collectives, exhibit the capacity for action, while macro-level entities do not. Coleman's meta-model was developed further by Hedström & Swedberg (1996), and we use their terminology in its introduction. Here the "agency level" explicitly refers either to individuals or groups of individuals acting collectively. *Situational mechanisms* (macro-micro) explain how individuals' internal states are impacted by exposure to social structures in their social worlds, i.e. the "situations" in which individuals are embedded. *Action mechanisms* (micro-micro) explain how individuals perform actions because of their internal states. *Transformational mechanisms* (micro-macro) explain how social structures are constructed and affected by the collective actions of individuals. As shown in Figure 1, the sequence of mechanisms, taken together, constitute a mechanism-based theory of social dynamics. This meta-model is a cornerstone of the analytical sociology tradition. Manzo (2007) systematically studied analytical sociology's trio of mechanisms drawing further on Coleman's work, and incorporated within the meta-model an explicit temporality with indications of how mechanisms could be chained together through time.

1.4   The Hedström & Swedberg (H&S) meta-model (along with the Coleman Boat and its derivatives) is a special case of the general micro-macro schema and takes the methodological individualist epistemological stance. Only individuals exhibit agency. Situational mechanisms generate social structure for individual-to-individual interactions, and all changes to social level entities must be brought about through the actions of individuals through a macro-micro-macro chain of mechanisms. Macro-macro mechanisms are discouraged (Ylikoski 2016).

Alternative meta-models

1.5   Tilly (2001) defined a three-part meta-model which does not take the methodological individualist position and distinguishes between environmental, cognitive and relational mechanisms. Environmental mechanisms are external influences affecting individuals (macro-micro) or social structures (macro-macro). Cognitive mechanisms specify individual behaviours (micro-micro). Relational mechanisms "alter connections among people, groups, and interpersonal networks" (micro-macro) (Tilly 2001), but may operate between social level entities (macro-macro). This scheme is similar to the Coleman Boat; however, Mayntz (2004) points out the important difference that relational mechanisms emphasise relations (structures) at the macro level, not only individuals at the micro level. In stark contrast to the H&S meta-model, both relational mechanisms and environmental mechanisms can affect social structures (e.g. political processes) "without any necessary connection to individual-level cognitive mechanisms" (Tilly 2001).

1.6   The Tilly meta-model, in allowing macro-macro mechanisms, supports the development of multi-layered mechanism-based theories: theories that have both processes that include individual action and processes that occur entirely at the social level. A further meta-model, developed by Sawyer (2005), has taken the multi-layer approach further: proposing a five-layer ontology comprising individuals, interactions, ephemeral emergents, stable emergents, and social structures. In the extension to five layers, Sawyer notes that the boundaries between layers become less distinct.

1.7   Archer (1995), taking a critical realist epistemological perspective, proposed a social morphogenetic approach containing three-part cycles: structural conditioning (macro-micro), social interaction (micro-micro), and structural elaboration (micro-macro). She argues that the interplay between individualism and collectivism is crucial for effective social theorising, rejecting the methodological individualist position. Macro-micro interaction can occur directly between social entities and individuals without requiring individual-to-individual interaction.

Macro level

Social entities or phenomena (e.g. institutions, regulations, norms) which also have attributes (e.g. scale, direction, features) which can change over time

Macro-macro mechanisms

Situational mechanisms

Transformational mechanisms

Action mechanisms

Micro level

Micro entities (e.g. people, household) who can interact, and have attributes (e.g. characteristics, internal states) which can change over time.

Note:

Situational mechanisms (macro → micro) explain how micro entities' internal states are impacted by exposure to social entities which have particular features, in their social worlds (i.e. the individuals are affected by the "situations" in which individuals are embedded).

Action mechanisms (micro → micro) explain how individuals perform actions because of their internal states.

Transformational mechanisms (micro → macro) explain how social structures are constructed and affected by collective actions of micro entities.

Macro-macro mechanisms (macro → macro) explain how social entities directly affect other social entities. We do not explicitly disallow macro-macro mechanisms in MBSSM architecture.
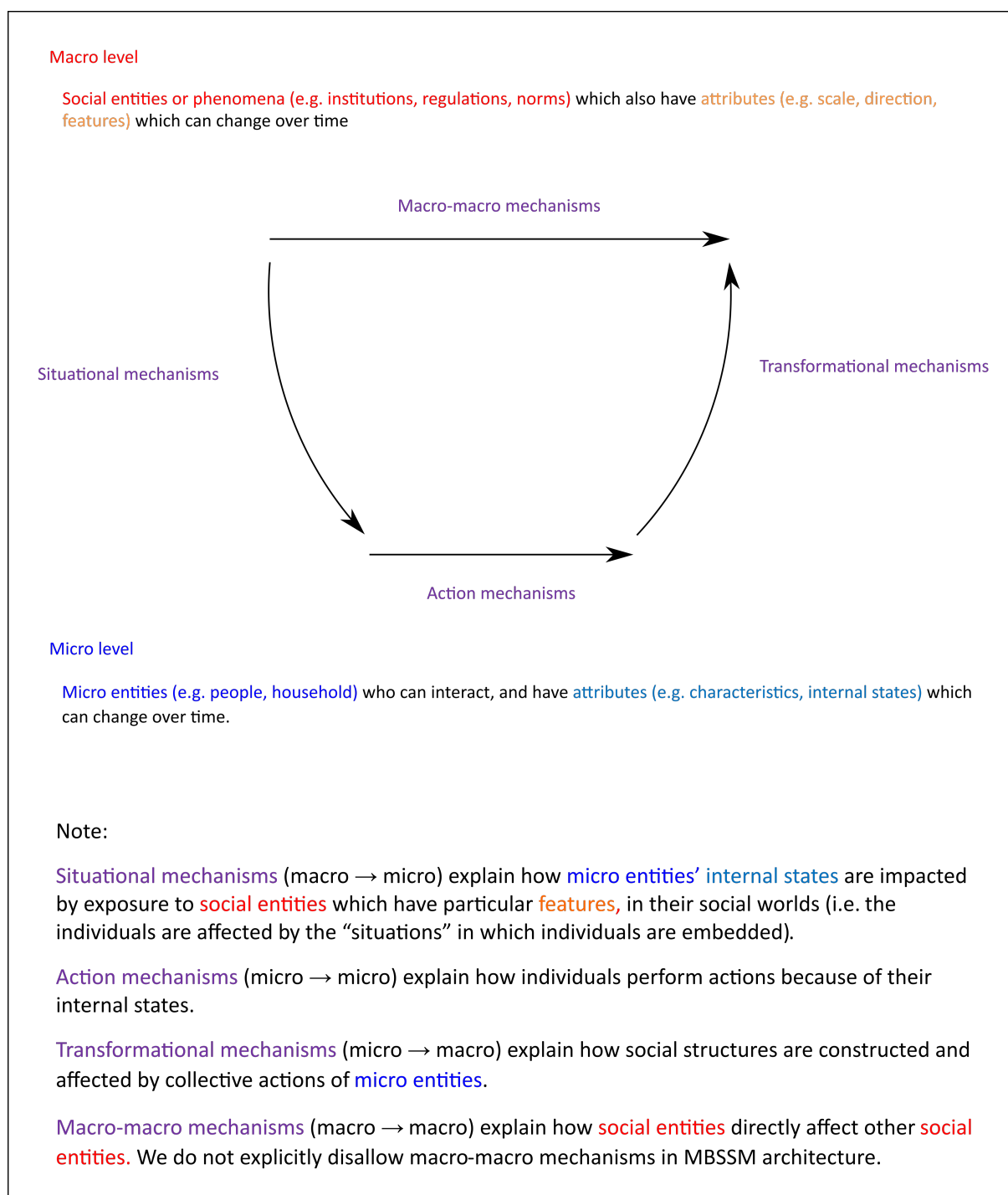
Figure 1: Key conceptual components in the Mechanism-based social system modelling (MBSSM) framework of Hedström & Swedberg (Hedström & Swedberg 1996)

1.8 There is a lively debate among scholars regarding the epistemological validity of the different meta-models. For example, Mayntz (2004) suggested a "built-in bias" of individual-focused action in the Coleman Boat and the subsequent H&S meta-model. Jepperson & Meyer (2011) criticise methodological individualism as an insufficient meta-theory to explain many social processes, which they argue are inherently multi-level.

The general micro-macro scheme

1.9 To provide the flexibility to construct models from different epistemological perspectives, we designed our architecture based on the general micro-macro scheme, with the three mechanism types (macro-micro, micro-

micro, macro-micro mechanisms) that appear in *all* of the meta-models discussed above. We adopt the terminology of the H&S meta-model (situational, action, transformational mechanisms in Figure 1) but we do not restrict the micro-level to consist of individuals, nor do we restrict interaction to occur between individuals only, nor do we explicitly disallow macro-macro mechanisms.

1.10   To illustrate the general micro-macro scheme, consider the well-known segregation model proposed by Schelling (1971). Schelling aimed to explain the stylised fact of neighbourhood segregation with a mechanism-based model. He used pennies and dimes to represent households of two different social groups and a chessboard to represent the city that these households reside in. Schelling revealed that even a low level of preference for neighbours of the same social group could lead to severe segregation over time. In Schelling's model, the micro level entities are households and the macro level entity is the city. The households, represented by the coins, have attributes including location, social group, and preference for their own social group. The city, represented by the chessboard, knows the location of all households, can provide households with information about their neighbouring households, and has its own attribute, the segregation index, which is a calculated measure of the spatial clustering by social group. At the outset households are distributed randomly in the city. This mechanism-based theory can be represented with the general micro-macro scheme as follows: (1) situational mechanism households observe the proportion of nearby households in the same social group; (2) action mechanism the households will choose to move if there are not enough households from the same social group, i.e. by comparing the proportion against their own preference for similar neighbours (e.g., "I am a dime; at least 30% of my neighbours must be dimes."), and move to a random location in the city if preferences are not met; (3) transformational mechanism in the aggregate, household movement alters the spatial distribution of households in the city and this rearrangement of spatial clustering can be quantified using the segregation index. As the chain of mechanisms are repeated for all households, the city changes in character and the segregation pattern can emerge over time. Schelling's seminal model of segregation gave fascinating insight into how households identifying with different social groups can self-segregate even when they have low preference for similar neighbours.

A mechanism-based social systems modelling (MBSSM) architecture: Benefits and requirements

1.11   Manzo (2007) assessed different types of modelling paradigms (statistical, computational, and mathematical modelling) for constructing generative sociological models and concluded that the computational simulation model is "an adequate infrastructure for the formalisation and, above all, for the analysis" of generative models. Since mechanism-based meta-models focus on mechanisms executed on and by actors, mechanism-based models are often represented effectively using an *agent-based computational model*. Agent-based modelling and simulation (ABMS) is a methodology in which the system is modelled from the bottom up by defining a collection of agents (comprising the micro level) and their interactions (Epstein 2006). In agent-based models, local agent interactions give rise to emergent behaviour at the macro level; thus, ABMS is a powerful tool to study and understand social systems, explain the underlying mechanisms, provide predictions, and potentially discover and formalise theories (Gilbert & Troitzsch 2005). Despite its focus on agents and local interactions, Marchionni & Ylikoski (2013) argue that ABMS is not restricted to implementing methodological individualism, but is well suited for generative mechanism-based modelling in the round.

1.12   However, it is not an easy task to operationalise a social system within an agent-based model. To date, mechanism-based explanations are often rich but hard to operationalise, whilst operational models tend not to explicitly emphasise mechanism-based explanations. Operational models that draw from several theories are typically ad hoc and open to criticism that they lack theoretical grounding. For example, Bourgais et al. (2018) surveyed emotional modelling in social simulation and gave examples of theory-based models at the two extremes:

- Zoumpoulaki et al. (2010) designed an evacuation simulation with the OCEAN model of personality and the OCC model of emotions. Emotional status represented by a vector of emotion types was generated in an agent using cognitive appraisal theories. The basis of Bourgais et al. (2018)'s criticism here was, "knowledge of emotions' theories required, complicated to implement".

- Le et al. (2012) represented intensity and duration of fear emotions with two numerical values and integrated the emotion into the model using reactive creation (i.e., the perceptions of given events create emotions, with no cognition involved). Bourgais et al. (2018) criticised the model as, "too simple to realistically cope with the process of emotions' creation, no support from a psychological theory".

1.13   Within a critical realist framework, AMBS can be considered as a tool for abductive reasoning – the scientific process of identifying the mechanisms that have generated a particular phenomenon (Miller 2015). This reasoning

process consists of two parts (Danermark et al. 2002): *redescription* (situating the concrete phenomenon as case which emerges from the hypothesised interacting components (i.e. entities and mechanisms) of one or more theories); and *retroduction* (identifying which of the components in the redescription are fundamentally constitutive to the emergence of the phenomenon). In ABMS, redescription produces a model, whilst retroduction involves running the model to see if it can generate the observed phenomenon in which case, the model becomes a candidate generative explanation (Epstein 1999). However, what is lacking here is a framework for searching across model structures to identify other potential candidates. Done by hand, this would be laborious work. Gunaratne & Garibay (2017) and Vu et al. (2019a) have recently undertaken such model discovery work using machine learning methods. However to enable these methods to be applied at scale requires a common language for expressing the diverse range of mechanisms and entities that constitute a theory. The notion of a common formal language was also suggested by León-Medina (2017) to facilitate analysis of mechanism-based simulation models.

1.14 To support dynamical modelling of mechanism-based social theories, we designed a software architecture called the Mechanism Based Social Systems Modelling (MBSSM) architecture. A software architecture is the high-level structure of a software system. It defines the elements of a software system, what those elements do, and the relations between them. The MBSSM software architecture is a realisation of the formalisation of the general micro-macro scheme. It provides a broad framework for mechanism-based explanation that can incorporate ABMS and allows flexibility for modellers to implement and explore different mechanistic pathways. Additionally, the MBSSM architecture should support multi-theory modelling, in which mechanistic pathways from multiple theories are incorporated in a unified model for better explanation. This is because a theory sometimes only explains a limited aspect of an observed phenomenon. Different mechanistic pathways from different theories may complement each other and enrich the overall mechanism-based explanation.

1.15 There are three functional requirements for the MBSSM architecture. Firstly, to be faithful to the general micro-macro scheme, it must model a social theory as synchronous or asynchronous sequences of macro-micro, micro-micro, micro-macro, and/or macro-macro mechanisms. Secondly, after social theories are conceptualised via social mechanisms in the general micro-macro scheme, the MBSSM architecture needs to support combinations of different theories for the same phenomenon and enable a search for hybrid theories that better explain emergent behaviour in the social system. Thirdly, the architecture must be extensible and maintainable (i.e., it is easy to add more functionality and easy to fix bugs). For this requirement, we followed the ABOOMS (Agent-Based Object-Oriented Modelling and Simulation) development framework (Vu et al. 2019b) to ensure the software engineering principle *separation of concerns*: separating computer software into sections such that each section addresses a separate concern and overlaps in functionality are reduced to a minimum.

1.16 To summarise the functional requirements, the MBSSM architecture must:

1. facilitate the implementation of individual mechanism-based social theories;

2. support combinations of several alternative theories for the same phenomenon;

3. be extensible and maintainable.

Aims of the paper

1.17 The paper aims to develop an MBSSM architecture to provide a natural path to implementation and simulation of mechanism-based social theories in an operational agent-based simulation model and also aid in mechanism-based theory development according to the general micro-macro scheme.

1.18 To meet this aim, the following objectives will be delivered in this paper:

1. Describe the design and implementation of the MBSSM architecture in the objective-oriented programming paradigm with Unified Modelling Language (UML) diagrams.

2. Demonstrate how to populate the MBSSM architecture with concepts and mechanisms from mechanism-based social theories to implement an agent-based simulation model.

3. Implement a multi-theory model and discuss how the MBSSM architecture supports the combination of multiple theories for theory exploration and development.

The MBSSM Architecture: Mechanism Typology, Software Design, Model Calibration

Situational mechanisms: Situated individual-to-individual interaction and structure-to-individual interaction

2.1   In the general micro-macro scheme, situational mechanisms encode interaction: where an individual's attributes are modified based on the actions or observable attributes of other entities. The entities being observed may be either individuals or social entities, or both, depending on the epistemological position adopted.

2.2   In ABMS, interaction focuses on agent-to-agent interaction. The 'macro' in the situational mechanism here relates to the relationship or connection between the agents, which is typically encoded as a network structure or spatial environment that suggests the existence of, and provides context to, the interaction. For example, an individual may change their beliefs about alcohol use in response to observations of the alcohol use behaviours or stated intentions of friends and family (i.e. a form of social contagion (Christakis & Fowler 2013; Steglich et al. 2010)). Interactions may also be defined between individuals and social entities, where an agent perceives a social structure that cannot adequately be represented via agent-to-agent interactions. For example, an individual may change their beliefs about alcohol use in response to perceived norms about alcohol use behaviours that are encoded in the wider socially constructed environment (e.g., the presence and nature of alcohol outlets, media messages related to drinking identities, or the political culture related to drinking (Jayne et al. 2008; Room et al. 2009; Scott et al. 2017) in addition to personal interactions. The MBSSM architecture can support explanations of this kind by distinguishing between micro and macro entities and allowing modellers to design the interface of these entities in order to allow restricted access to the entities' observable attributes.

Action mechanisms: Conscious and unconscious decision making by individuals and developmental processes

2.3   In the general micro-macro scheme, action mechanisms encode a process by which an individual makes a decision, either consciously or unconsciously, based on its own attributes and subsequently performs an action or behaviour based on that decision. Using the folk psychology often used as an exemplar mechanism in analytical sociology, the agent's action is a function of its beliefs, desires and opportunities that pertain to the decision. Other types of action mechanism are possible: for example, in the Theory of Planned Behaviour (Ajzen 1991), the intention to perform an action is based on attitudes, perceptions of norms, and perceptions of behavioural controls. The MBSSM architecture also supports within-agent developmental processes (e.g., addiction, reinforcement) being encapsulated within an action mechanism. For example, in the Schelling example, the modeller might wish to represent an intra-agent neurobiological process (e.g., memory consolidation or habit formation) in which a household's threshold (i.e., tolerance) for moving is also based on how long it has stayed in the same place. In MBSSM terms, this process is a micro-micro mechanism in which an agent's behaviour subsequently affects that agent's attributes. If the modeller wishes the effect to occur instantaneously with the behaviour, the process can be included within the action mechanism that generates the behaviour. If the modeller wishes the effect to accrue *after* the behaviour has occurred, then past behaviour is encoded as an attribute of the agent and a new micro-micro mechanism is defined to perform the feedback effect, making use of this attribute.

Transformational mechanisms: Simple and complex aggregation

2.4   The definition of transformational mechanisms can be regarded as one of the most challenging and, arguably (León-Medina 2017; Ramström 2018), problematic aspects of realising the general micro-macro scheme a point articulated by Coleman during his original treatise on the Boat (Coleman 1986).

2.5   Transformational mechanisms, when considered separately from situational mechanisms and action mechanisms, must always involve some kind of aggregation of actions from multiple micro-level entities to either replace or adjust one or more states of a macro-level entity. It is important to recognise that aggregation is, in general, not the same as summation of individual actions at a single point in time. The aggregation may be synchronous (as in many ABMS) or asynchronous (as in, for example, the stochastic actor based model — SABM Steglich et al. 2010). The aggregation may also be mediated by other macro-level and micro-level entities (e.g., the transformation mechanism by which individual votes (i.e., actions) are aggregated to generate a government is mediated by separate entities: the (macro-level) institutional rules that define the way in which votes

are translated to representatives and the (micro-level) individuals who enforce those rules. The aggregation may also adjust rather than replace an existing structure – e.g., the election may only have covered a certain proportion of representatives and so the existing state of the government is important in determining its revised form.

2.6 Whilst transformation mechanisms are about aggregation, this does not, in general, limit the ability of the overall model to display complex, emergent behaviour. Manzo (2007) usefully distinguished between simple aggregation mechanisms and complex aggregation mechanisms. In simple aggregation mechanisms, the aggregation is based on acting agents whose states (e.g., beliefs, desires, or opportunities) are not affected by the states or actions of any other agents at any point in time. In the general micro-macro scheme, this type of simple aggregation would be very rare. This point becomes clear when we consider the nature of Manzo's complex aggregation mechanisms. Manzo distinguishes between two types of complex aggregation: direct and indirect. In the former, the aggregation of actions is based on acting agents whose states have been subject to interactions with other agents (i.e., situational mechanisms). In MBSSM, direct complex aggregation is achieved by sequencing these individual-to-individual situational mechanisms prior to the transformation mechanism that does the aggregating. In Manzo's indirect form of complex aggregation, the aggregation of actions is based on acting agents whose states have been subject to interactions with macro-level entities (i.e., situational mechanisms, but not those involving individual-to-individual interactions). Therefore, in MBSSM, indirect complex aggregation is also achieved by sequencing situational mechanisms prior to the transformation mechanism. When considering the totality of the general micro-macro scheme, and its MBSSM implementation, it is apparent that sequencing of situational and/or action mechanisms prior to the triggering of a transformational mechanism will produce indirect and/or direct complex aggregation or some hybrid of the two.

Concepts in the object-oriented programming paradigm

2.7 Because ABMS is a natural choice for mechanism-based social modelling, we chose to develop the MBSSM architecture using the object-oriented programming paradigm, in which the basic unit of construction, an *object*, maps well onto the ABMS concept of an *agent*. An agent in ABMS has heterogeneous characteristics and can make decisions or perform actions. To aid understanding for readers not familiar with the object-oriented programming, we define several key terms related to the paradigm:

- **Object**: An object is the basic unit of construction of the object-oriented system that has different *attributes* and can perform *operations*. An operation can be a simple action or a complex collection of actions and other operations. For example, John's white car is an object, which has two attributes: its colour is white and its owner is John, and the car has an operation: drive forward.

- **Class**: Objects of the same type can be grouped into a class. A class is a template consisting of attributes and operations for creating objects. For example, John's white car and Jane's black car belong to the class Car. Class Car has two attributes (colour, owner) and an operation (drive forward).

- **Object-Oriented Analysis and Design (OOAD)**: The process of breaking down a complex system into various objects by identifying requirements and developing a design of an object-oriented software system that satisfies those requirements.

- **Unified Modelling Language (UML)** (Fowler et al. 2004) is a widely used graphical language to develop and document object-oriented systems. It is flexible and independent of OOAD process and implementation software. The importance of using UML for ABMS in social science has been stressed in (Bersini 2011; Collins et al. 2015).

- **UML Class Diagram**: A class diagram is a UML diagram for representing the structural design of the system by listing the classes, their attributes and operations, and the relationship between these classes. A class is represented by a box that consists of three parts: (1) the name of the class; (2) attributes of that class; and (3) operations that a class can perform. Relationships between classes in a class diagram:

  1. **Association** n declares that there is a link between two classes A and B. A bi-directional association (A and B know each other) is indicated by a solid line between two classes. A uni-directional association (only one of A and B know about the other) is indicated by a solid line with an open arrowhead. The arrowhead points to the class that does not know about the association.

  2. **Generalization** expresses an inheritance relationship between a more general class (*superclass*) and a more specific class (*subclass* or *derived class*). The subclass inherits attributes and operations of

its superclass, but can still have its own attributes and operations. The subclass can also override the superclass' operations with its own implementation. The relationship is indicated by a solid line with a closed arrowhead. The arrowhead points to the superclass and the opposite end connects to the subclass.

- **Abstract class**: An abstract class cannot be used to create objects but can be derived or subclassed. The purpose of this type of class is to serve as a template for other classes.

2.8 Manzo (2014) discussed why and how object-oriented programming is an asset for mechanism-based modellers. Firstly, because an object is conceptually empty, agents (built from objects) can be constructed with any type of action theories, from neoclassical rational-choice theory to complex cognitive theories. Secondly, due to conceptual emptiness, agents can represent not only real-world actors like individuals, but also different types of entities at different conceptual levels such as organisations at the macro level. Objects can also be used to construct spaces and networks at the macro level, which are essential for agents' interaction. Lastly, the object-oriented programming can naturally encode social heterogeneity: different object classes can have different attributes, objects of the same class can have different values for an attribute or different behaviours in operations, and objects can be scheduled to act at different points in time. Our framework leverages the structural homology between ABMS methodology and real-life mechanism-based systems to design computational generative models.

Object-oriented analysis and design for the MBSSM architecture

2.9 The OOAD process was performed to translate the requirements of enforcing the general micro-macro scheme (Figure 1) into the classes required in the MBSSM architecture. The complete structural design of the MBSSM architecture is presented in the UML Class Diagram (Figure 2). Our design includes seven classes:

1. `MicroAgent`: represents a low level entity at the micro level.

2. `StructuralEntity`: represents a social structural entity at the macro level, such as institutions, regulations, norms, etc.

3. `Model`: creates/deletes agents and structural entities, manages the events, facilitates the interaction between agents and social structural entities, and collects simulation results.

4. `Theory`: in charge of situation and action mechanisms of a theory for the Agent class.

5. `TheoryMediator`: combines situation and action mechanisms from different Theory classes.

6. `Regulator`: in charge of transformational mechanisms for the StructuralEntity class.

7. `StructuralMediator`: combines transformational mechanisms from different StructuralEntity classes.

2.10 In the class diagram (Figure 2), the basic classes of the simulation are `Model`, `MicroAgent`, and `StructuralEntity`. For example, the `MicroAgent` class may represent a person (or other low level entities like households). The `MicroAgent` class has operations: `callSituation()` and `callAction()` which trigger situational mechanisms (reacting to the social environment by changing internal states) and action mechanisms (acting due to internal states). The `Model` class has the operations for creating `MicroAgent` objects in the simulation, managing scheduling events during a simulation run, and collecting simulation outputs such as agent statistics. The `StructuralEntity` class represents a macro-level social entity of the social system and trigger transformational mechanisms via `callTransformation()` operation, i.e. update structural entities attributes due to collective actions of individual agents. A benefit of object-oriented programming is that it allows the attributes of an object to be public (observable to all) or private (only known to the object itself). Therefore, modellers can restrict which attributes an object exposes to other objects.

2.11 To facilitate the combination of theories, the architecture uses a design approach called the *mediator design pattern* (Gamma 1995). The mediator design pattern is applied to micro-level interactions (`MicroAgent`, `Theory`, `TheoryMediator`) and macro-level interactions (`StructuralEntity`, `StructuralMediator`, `Regulator`). This design decision originated from the need to include multiple theories for the same phenomenon. Let us discuss the design evolution by dissecting the action mechanisms of agents at the micro level. The simplistic approach is to include all theories in the `MicroAgent` class by adding attributes and operations related to each specific theory. This design is difficult to manage, since the number of attributes and operations within the `MicroAgent`
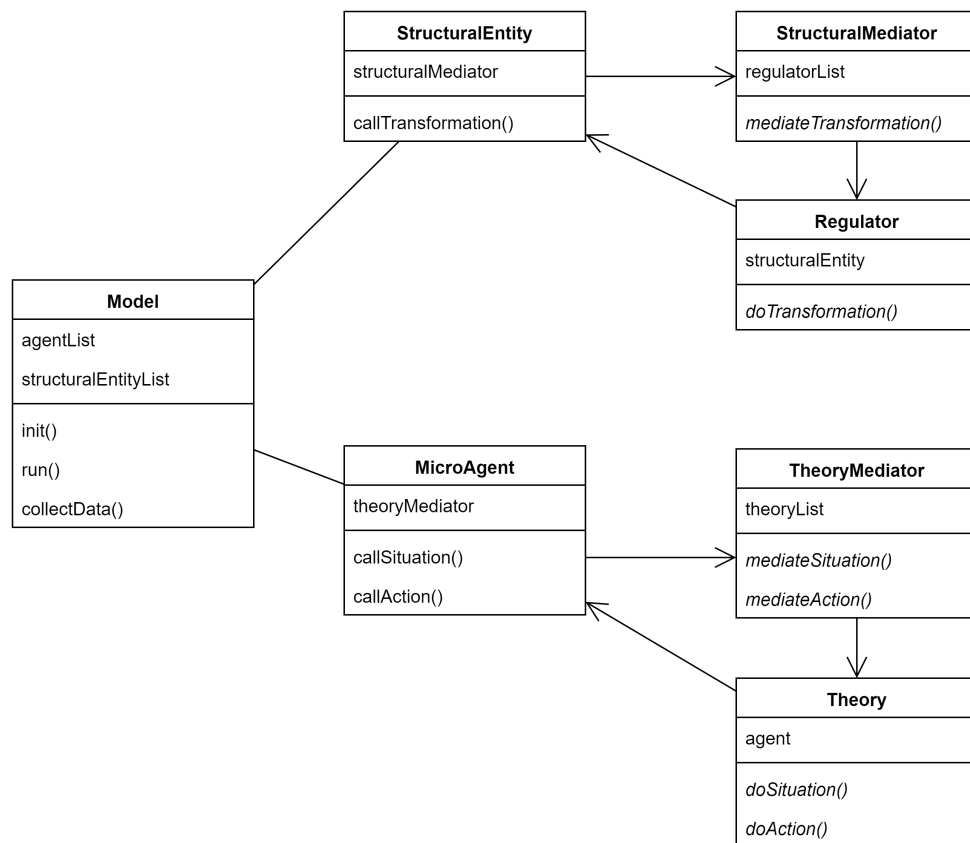
Figure 2: Class diagram Generic version for mechanism-based social systems modelling. (Note: The attribute names end with "list" means that it is a list of objects of a certain class. For example, since `Model` manages all `MicroAgent` and `StructuralEntity` objects, it has two lists of those objects `agentList` and `structuralEntityList`. Operations in italic are abstract operations, which mean they do not have an implementation thus the derived classes must override these operations.)

class increases as more theories are added to the model. Another design approach is to separate attributes and operations into two groups: theory-specific ones in the `Theory` class and the common ones that can be shared between theories in the `MicroAgent` class. However, if multiple `Theory` objects connect directly to a `MicroAgent` object, they can have conflicts when influencing the `MicroAgent` behaviour, some exerting upwards and some exerting downward pressure on an agent variable. Another concern is that some phenomena cannot be explained by a single theory but the combination of multiple theories. Implementing a synthesis or combination of theories in `MicroAgent` or `Theory` objects can result in many interconnections between these objects and would require each theory object to know about the existence and operations of all other theories. When a new theory class is added, there is implementation required in all existing theory classes. The solution for this problem is the mediator design pattern, in which the theory combination is encapsulated in a separate mediator object, `TheoryMediator`, between a `MicroAgent` object and other `Theory` objects. `TheoryMediator` object has the responsibility of controlling and coordinating interaction between `Theory` objects. It keeps the `Theory` objects from referring to each other explicitly. A theory only knows the mediator and does not need to know other theories. With this design, the mechanism-based theories can be implemented separately then combined in various ways via the mediator. The combination can be done manually or automatically to explore different mechanism combinations.

2.12   The situational and action mechanisms are carried out by the collaboration between three classes: `MicroAgent`, `TheoryMediator`, and `Theory`. A `MicroAgent` object triggers these two mechanisms by invoking `callSituation()` and `callAction()` operations. These operations will invoke `mediateSituation()` and `mediateAction()` operations of the `TheoryMediator` object. `TheoryMediator` then invokes `doSituation()` and `doAction()` in `Theory` objects, and combines the relevant information from `Theory` objects to inform the `MicroAgent` object. Similarly, the transformational mechanisms are performed by a series of function calls between three classes: `StructuralEntity`, `StructuralMediator`, and `Regulator`. It starts from `callTransformation()` in `StructuralEntity`, to `mediateTransformation()` in `StructuralMediator`, and to `doTransformation()`

in `Regulator`. The mediator pattern supports the theory combination at both micro and macro level.

2.13 The four classes, `TheoryMediator`, `StructuralMediator`, `Theory`, and `StructuralEntity`, are abstract classes. This means that the classes and their methods do not have any implementation. When modellers create a new theory, they need to provide the implementation for the methods in these classes. Modellers have to provide the implementation for mechanisms via `doSituation()`, `doAction()`, `doTransformation()`, as well as the approach for combination via `mediateSituation()`, `mediateAction()`, `mediateTransformation()`.

2.14 In summary, the software architecture is documented in a UML class diagram and supports the three mechanism types of the general micro-macro scheme. Additionally, the architecture facilitates the implementation of multiple mechanism-based theories with the mediator design pattern. Core classes are identified and can be extended by modellers for their model development. In Section The MBSSM Architecture in Practice, we apply the software architecture to model a mechanism-based theory.

Software implementation

2.15 The MBSSM architecture can be implemented in any object-oriented language or simulation toolkit that supports object-oriented programming. The implementation process will be different depending on the features supported by the software. In this paper, RepastHPC (Collier & North 2013) was chosen for model implementation. RepastHPC uses the C++ language and supports parallelism on high performance computing. Using a simulation toolkit has the benefit of not requiring that modellers build the basic components of a simulation such as a scheduler or an agent network. However, modellers have to follow certain procedures set out by the simulation toolkit. For example, in RepastHPC, classes of any agent type must have an AgentId attribute that is defined by RepastHPC. Thus, when implemented in RepastHPC, the AgentId attribute has to be added to `MicroAgent` class. Similarly, the MBSSM's Model class has to follow RepastHPC procedures to set up agents, schedule events, etc. The RepastHPC implementation of the MBSSM architecture is available to download via `https://bitbucket.org/r01cascade/MBSSM_architecture` and is licensed under the GNU General Public License version 3.

Model calibration with the MBSSM architecture

2.16 Models of social systems are based on our understanding of the real world. We can observe the emergent patterns but are unlikely to be able to observe directly many aspects of the system, such as the decision making processes of individuals. Calibration refers to the process whereby we measure how well a model matches observed data for a range of parameter values and/or structural choices. Parameter values and structural choices that lead to a close fit of the model to real data are given high weight, whereas parameter values and structural choices that lead to a poor fit are given low weight.

Parameter calibration

2.17 We recommend a probabilistic approach to parameter calibration, in which our knowledge about each parameter is represented using a probability distribution. Before calibrating the model, we have prior beliefs about the true values of the parameters, which we express as probability density functions. We also have historical data concerning the phenomena of interest that the model has been built to predict. This historical data is used as a *target* for calibrating the model parameters. The calibration process results in a posterior distribution for each parameter, which represents a synthesis of the prior beliefs with the information in the historical target data. We use an Approximate Bayesian Calibration (ABC) approach (van der Vaart et al. 2015) to generate posterior beliefs for the parameters the posteriors can then be used in explanatory discussion and also for predictive purposes. The application of Bayesian approach to parameter calibration of complex models is increasing in many fields such as agent-based modelling, ecology, and epidemiology (Beaumont 2010; Grazzini et al. 2017; Lenormand et al. 2013).

2.18 Non-probabilistic approaches to parameter calibration have also been proposed, and represent the most common approach for calibration of agent-based models; for a good review, see Thiele et al. (2014). For example, Purshouse et al. (2014) used an evolutionary optimiser to identify parameterisations of a theory-driven simulation of alcohol consumption dynamics in England between 2003 and 2010. As another example, Stonedahl & Wilensky (2011) utilised BehaviorSearch a software tool that uses evolutionary search algorithms to search a model's parameter space to search for parameters that can produce different flocking behaviours. The approach involved the systematic confrontation of the simulated outputs with the targeted "reference pattern"

to be explained. This pattern validation approach modifies the parameters related to behaviour of the agents: birds' minimum distance, flying speed, acceleration in the flocking models.

Structural calibration

2.19 Additionally, it is also interesting, both for theory development and also when using models for prospective policy appraisal, to explore the *structure* of the model to see which combination of mechanisms or theories leads to the best fit between the model predictions and observed data. This process is known variously as *structural calibration*, *model discovery*, and *inverse generative social science* (Vu et al. 2019a, 2020). It is important to differentiate between structural calibration and parameter calibration. Considering again Stonedahl & Wilensky (2011) flocking model (an example of parameter calibration), structural calibration takes things further than the current BehaviorSearch implementation: modellers can modify the agent decision making process altering if-else statements, the order of decision making, the structure of decision equations, or utility functions. In this way, structural calibration can be seen as an extension to (Grimm et al. 2005)'s pattern-oriented modelling approach for contrasting alternative theories, where alternatives now also include different representations of the same theory.

2.20 Structural calibration can be applied to mechanisms within a theory or a combination of multiple theories. A combination of theoretical components of multiple theories may provide a better understanding or explanation of a phenomenon than an individual theory. The mediator design in the MBSSM architecture provides a natural point where different combinations can be tested in the mediator class. The mediator separates the mechanisms (in the `Theory` and `Regulator` classes) from the affected entity (`MicroAgent` and `StructuralEntity` classes). Therefore, the modeller can add more mechanisms into the model without altering the agents and social structures. Furthermore, the mediator can be implemented with different combination techniques without affecting the theoretical mechanisms. This separation of concerns allows the modellers to be robust and flexible when working with multiple theories and combination techniques. Methods and tools for structural calibration are in their infancy; we return to this point in Section Discussion.

The MBSSM Architecture in Practice — A Worked Example of Social Norms Theory for Explaining Population-Level Alcohol Use Patterns

The case study of alcohol modelling with social norms theory

3.1 In this section, we demonstrate how to populate the MBSSM architecture by showing the development of a simulation of a single mechanism-based theory that aims to explain long term changes in population alcohol use. The question being asked is: which social mechanisms can explain the dynamics of alcohol use patterns observed within a defined spatiotemporal setting (e.g., California in the 1980s and '90s). The number of drinks of an individual on a given day would be an attribute of DrinkingAgent class and the DrinkingAgent drinking action would be carried out in the doAction() operation, returning the number of drinks. There are several candidate mechanism-based theories for why a person would drink a given amount in a day: *social norms* (a person may consider a given number of drinks appropriate based on social norms of drinking for people like himself/herself) (Rimal & Real 2005); *social roles* (a person may have little time to drink because of the number of social roles he/she holds, or may drink at a certain level to cope with the stress of holding social roles) (Kuntsche et al. 2009); *social contagion* (a person may be influenced to drink more or less by his/her friends, or may choose friends who drink similarly to himself/herself) (Steglich et al. 2010); or *rational choice* (a person may choose to drink a given amount by weighing up alternative uses for his/her time and money) (Becker & Murphy 1988). The case study in this chapter aims to use the theory of normative social behaviour (Rimal & Real 2005) to explain population alcohol use "norm theory" for short. We express the mechanisms in norms theory and develop an agent-based simulation using the MBSSM architecture.

Microsimulation

3.2 In order to operationalise a plausible explanation for observed drinking trends through time, we first need to adequately represent the population demographics in a way that is not necessarily directly linked to the social mechanisms under investigation. Therefore, before examining the social norms theory, we implement a

key component for the alcohol use model in this case study: a *microsimulation* of a synthetic population with plausible demographics and drinking-related variables over time. People with different demographic characteristics drink differently and the reference groups for norms are also split by demographic characteristics. The synthetic population (agents with representative socio-demographic attributes and drinking variables) can be constructed using individual-level survey and census data from the country that this model is interested in (Lomax & Norman 2016). We extended the `Model` class to create a new class, namely `MicrosimModel`, in charge of microsimulation functionalities: agent initiation with plausible demographics and drinking behaviour; the dynamics of ageing agents, births, deaths, and migration. This example of operationalisation is specific to our case study of alcohol use. The individuals do not have to be people; for instance, the microsimulation can also be remodelled to represent the drinking behaviour of households.

Alcohol modelling with social norms theory

3.3 In the context of alcohol use, norm theory postulates that people's attitudes about drinking alcohol are informed by the following concepts:

- **Descriptive norms** (Cialdini et al. 1991) which is the perception of how much other people drink. So the descriptive norms are the observation of how much people *actually* drink. An example mechanism is when more people drink, drinking is more visible and the descriptive norm results in liberalised attitude about drinking.

- **Injunctive norms** (Cialdini et al. 1991) which is the perception of acceptable or unacceptable drinking behaviour of other people, i.e. how much people *ought* to drink. For example, the norm could be women should drink less than men, or pregnant women should be abstinent from alcohol. Social situations can change the injunctive norm: For example, if a rise in the prevalence of heavy drinking leads to increased social disorder, drinking may be seen as less acceptable to society, resulting in a tightening of injunctive drinking norms (Holder, 1998).

- **Desire to drink** (Verhagen 2001) is the intrinsic desire to drink of a particular individual.

- **Autonomy** (Verhagen 2001) is the individual's tendency not to comply with the social norms. For instance, a person with high autonomy can follow their desire to drink while ignoring the norms (e.g. a person may drink one drink despite other people drinking three drinks in the same situation)

3.4 Not all norms apply to all people. The structure of normative influence is provided by a final concept: reference groups (Neighbors et al. 2008). People that have the same demographic characteristics are in the same reference group and have more influence on each other. For example, considering two demographic characteristics: sex and age group, a female teenager will be influenced most by other female teenagers. There is some influence from people with only one shared characteristic: women or teenagers. People with no shared characteristics have no influence at all (such as elderly men, in this example).

Functionality requirements for a social norms model

3.5 In this section, the social norms theory described in Section 3.1-3.4 was conceptualised to fit with the design of the MBSSM architecture. Further key components required for the norms model are also discussed in this section.

3.6 Conceptualising the social norms theory in the scheme (Figure /reffig3), we identified entities at micro and macro level as well as the three types of mechanisms.

- At the macro level, two *social structural entities* are injunctive and descriptive norms. Within each norm entity, the norms are separated into different reference groups (split by sex and age group).

- At the micro level, there are agents that represent the *individuals* that carry out the drinking behaviour.

- In *action mechanisms*, the drinking behaviour is influenced by four attributes of each person: perceived injunctive norm, perceived descriptive norm, desire to drink, autonomy.

- For *transformational mechanisms*, collecting drinking behaviour of individuals change the descriptive and injunctive norms. The descriptive norms are simply updated to summarise how people actually drink. For the injunctive norms, there are two mechanisms relaxation toward a more liberal descriptive norm and tightening due to societal concern that tend to pull the injunctive norms in opposite directions.
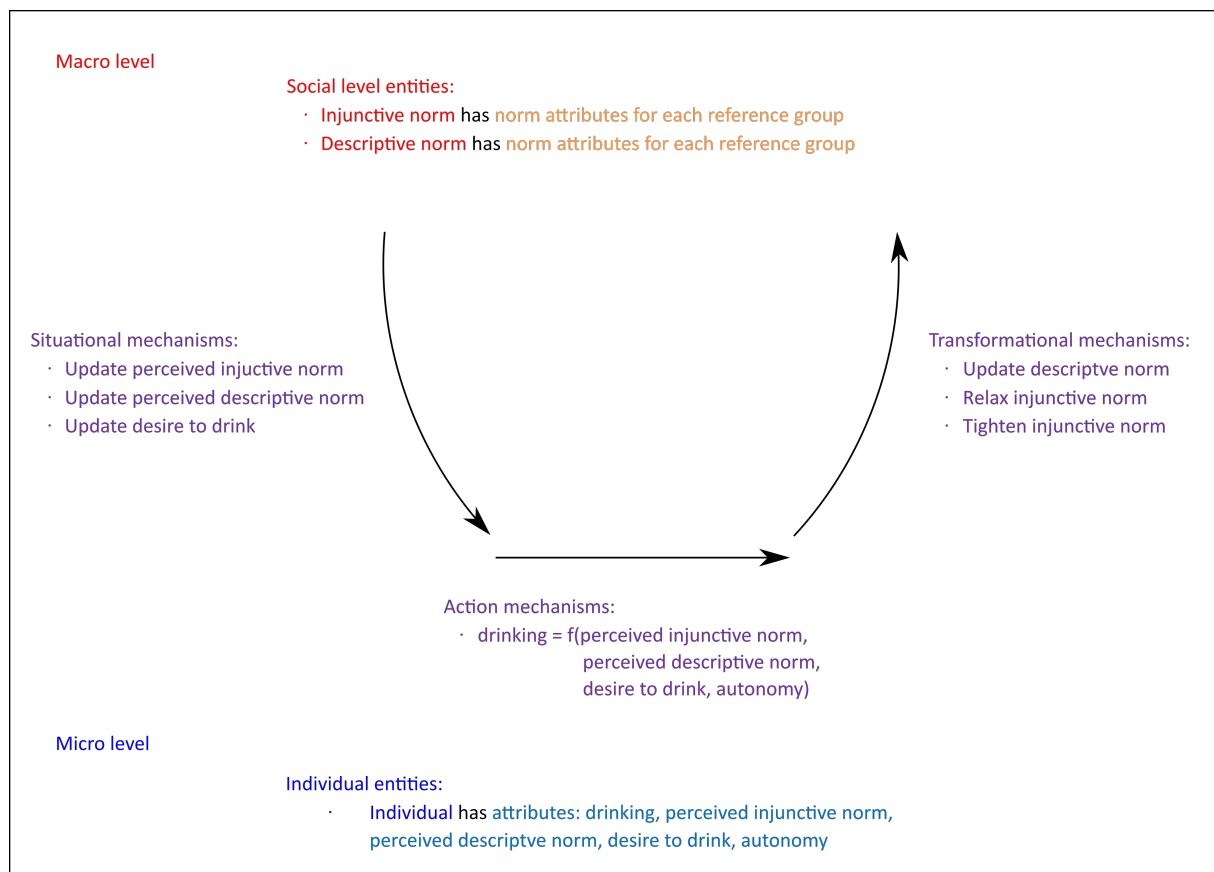
Figure 3: Social norms theory conceptualised in the general micro-macro scheme

- Finally, there are three *situational mechanisms* corresponding to updating the three attributes of each individual: perceived injunctive norm, perceived descriptive norm, and desire to drink.

3.7 To test whether norm theory as we have operationalised is a plausible explanation for observed drinking trends through time, we calibrate the model to empirical data over time. There are data requirements for the calibration process. To be calibrated, a model needs to read parameters and inputs from input files and then supply output data. The calibration process provides a setting to the model via the inputs and uses model outputs to evaluate model performance in that setting. Each input value for model calibration is systematically sampled from a probability density function which describe prior beliefs about the likely values of each model parameter. In this case study, the targets are drinking behaviour of different groups at the population level. The case study can be used to explore whether the norms theory can explain drinking trends observed in these groups, such as the convergence of male and female drinking in the US from 1979 to 2016 (Probst et al. 2020a).

OOAD and implementation of the social norms model

3.8 Based on this conceptualisation of norm theory, we undertook OOAD and extended the MBSSM classes in Figure 2. The resulting class diagram for the norms model is shown in Figure 4. For simplification, the MBSSM classes shown in Figure 2 were collapsed to omit attributes and operations to show only the class name and were grouped together into a core package in the centre rectangle of Figure 4. The new classes required for the norms model were added by *inheritance*. All the new classes are derived from the classes in the core package.

- `DrinkingAgent`: is derived from `MicroAgent` to include drinking-related attributes and operations. For example, the `drinkingHistory` attribute is the number of drinks the agent had each day in the past year. The `drink()` operation determines how much the agent will drink.

- `NormModel`: (is derived from `MicrosimModel`) initialises agents with the theory-related objects, initialises structural entities, controls the model during the simulation, and collects simulated data.
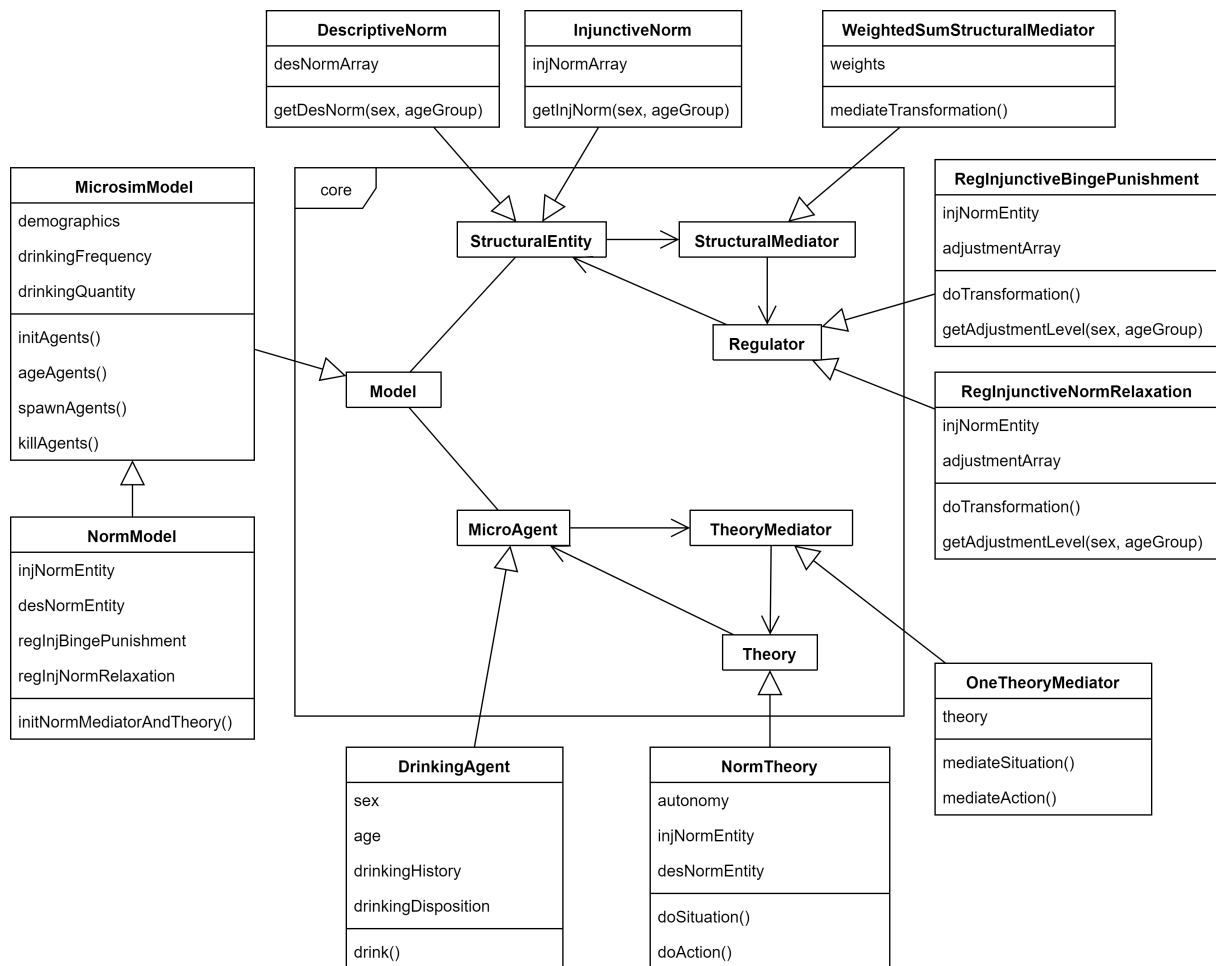
DescriptiveNorm

desNormArray

getDesNorm(sex, ageGroup)

InjunctiveNorm

injNormArray

getInjNorm(sex, ageGroup)

WeightedSumStructuralMediator

weights

mediateTransformation()

MicrosimModel

demographics

drinkingFrequency

drinkingQuantity

initAgents()

ageAgents()

spawnAgents()

killAgents()

core

StructuralEntity

StructuralMediator

RegInjunctiveBingePunishment

injNormEntity

adjustmentArray

doTransformation()

getAdjustmentLevel(sex, ageGroup)

Regulator

Model

RegInjunctiveNormRelaxation

injNormEntity

adjustmentArray

doTransformation()

getAdjustmentLevel(sex, ageGroup)

NormModel

injNormEntity

desNormEntity

regInjBingePunishment

regInjNormRelaxation

initNormMediatorAndTheory()

MicroAgent

TheoryMediator

Theory

OneTheoryMediator

theory

mediateSituation()

mediateAction()

DrinkingAgent

sex

age

drinkingHistory

drinkingDisposition

drink()

NormTheory

autonomy

injNormEntity

desNormEntity

doSituation()

doAction()

Figure 4: Class diagram for a norms model

- `NormTheory`: contains attributes needed for an agent to determine the number of drinks to have in a day according to norms theory: desire, perception of the injunctive and descriptive norms, and autonomy variables. `NormTheory` class also implements the `doSituation()` and `doAction()` operations, as required for the situational and action mechanisms.

- `OneTheoryMediator`: since there is only one theory, this mediator's responsibility is simply to pass the information from `NormTheory` to `DrinkingAgent`.

- `InjuctiveNorm`: represents the acceptability of drinking.

- `DescriptiveNorm`: represents the aggregated behaviour of other people.

- `RegInjunctiveBingePunishment`: refers to the reaction of the agent society to high levels of heavy drinking. The class determines the agents' level of drinking by reference group, compares to the injunctive norms for each reference group, and if needed makes the injunctive norms more restrictive.

- `RegInjunctiveNormRelaxation`: represents the adjustment of norms over time as a reaction to observed drinking behaviour in groups of agents. The class maintains observed drinking of reference groups in the agent population and adjusts injunctive norms.

- `WeightedSumStructuralMediator`: performs weighted sum of the adjustment values from the two regulators to determine the updated value for injunctive norms.

3.9 The norms model was implemented by extending the RepastHPC implementation of the MBSSM architecture.

Results of the social norms model

3.10 The social norms model described above is published in Probst et al. (2020b). In the paper, the model was used to explore the normative underpinnings of drinking behaviour in the US from 1979 to 1999. After calibration, the model was able to predict theoretically plausible changes in population-level drinking patterns via proposed normative mechanisms. Then the model was used to investigate three hypothetical policy scenarios. Experiments revealed that normative interventions are likely to have limited impact, given the empirically-informed importance of desire and autonomy to drinking behaviours.

Multi-Theory Modelling — Combining Social Norms Theory with Social Roles Theory

4.1 This section demonstrates how the MBSSM architecture can facilitate multi-theory modelling. We chose the case study of explaining population alcohol use with a combination of social norms and social roles theory. The design of the norms models in Sections 3.8-3.9 was extended to include role theory.

The social roles theory

4.2 A small number of studies have directly explored social role-based mechanisms that relate to the dynamics of population alcohol use (Knibbe et al. 1987; Wilsnack & Cheloha 1987; Yamaguchi & Kandel 1985). Between them, these studies examine and discuss how drinking behaviours are affected by the following role concepts:

- **Role status**: the holding of roles considered relevant to alcohol use behaviours: a marital role, an employment role, and a parenting role.

- **Role expectancy**: societal expectancy for an individual to hold a certain role based on that individual's perceived reference group.

- **Role overload**: the stress to an individual caused by holding combinations of roles.

- **Role incongruence**: the stress caused by the difference between an individual's current role status and societal role expectancy.

- **Role skill**: the ability to perform the role and avoid stress.

- **Role strain**: the overall stress that is caused by role overload and incongruence, eased by role skill.

- **Role selection**: heavy drinking makes it less likely that an individual will enter a role and also more likely that an individual will leave a role, due to the incompatibility of heavy drinking with role performance.

- **Role socialisation**: the longer an individual holds social roles, the more an individual's desire to drink is reduced.

- **Drinking opportunity**: the chances of being able to engage in a drinking occasion.

4.3 There is a complex interaction among these concepts. For example, the number of social roles a person holds may either restrict the opportunities for drinking or cause a person to drink more to cope with the stress. Mechanisms derived from these concepts are discussed in the next section.

Functionality requirements for a social roles model

4.4 To conceptualise social roles theory in the general micro-macro scheme, we identified entities at the micro and macro level as well as the three types of mechanisms (Figure 5). At the micro level, there are *individuals* who have role status and can perform drinking actions. At the macro level, there are two structural entities: *Role entity* that contains the role expectancy for each reference group (by sex and age group) and *Model* that contains the societal transition rates, which is an added microsimulation feature.

4.5 The situational mechanisms update agent variables as follows:

- **Role status**: three roles (marriage, employment, parenting) are updated via the role selection process. Agents transition in and out of their roles based on the societal transition rates stored in the Model structural entity.
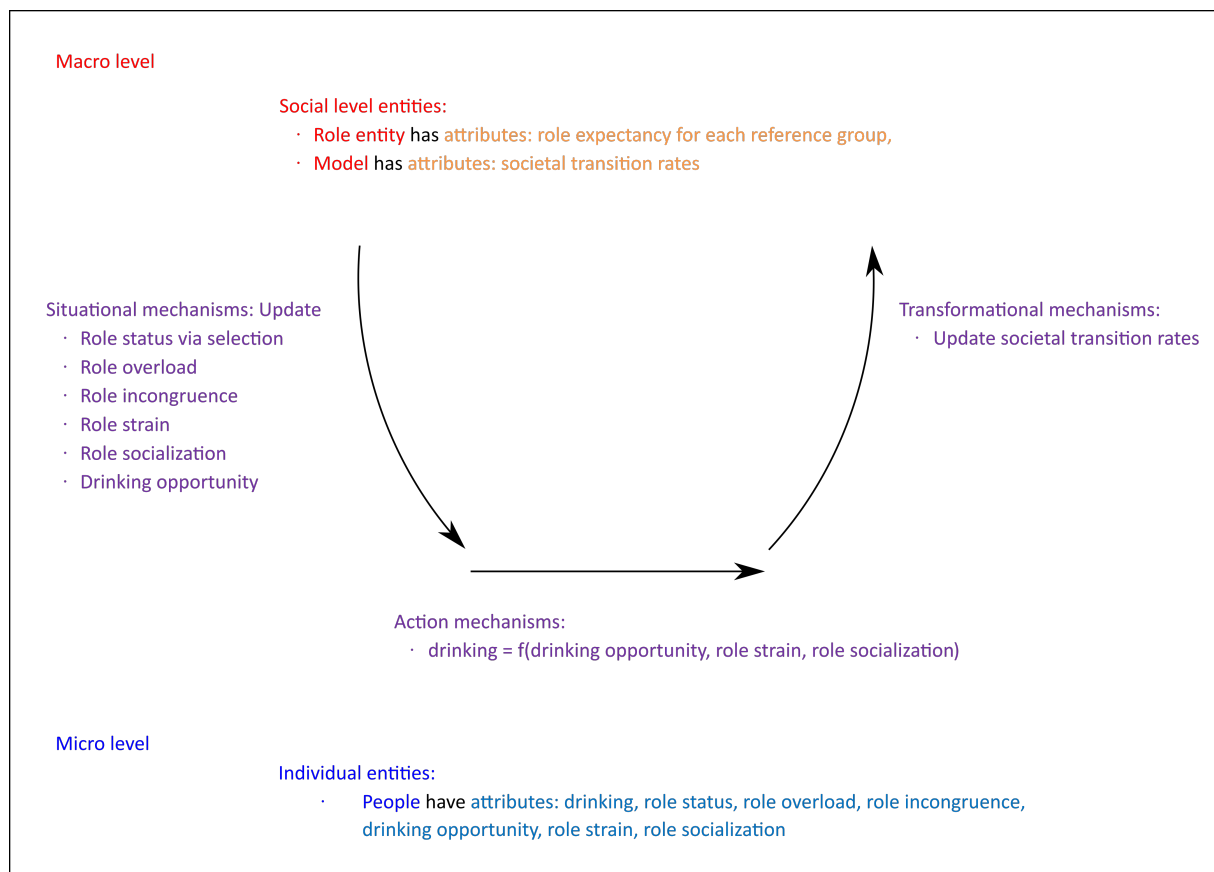
Figure 5: Social roles theory conceptualised in the general micro-macro scheme

- **Role overload**: a function of role status and the level of individual involvement for each role is updated as roles change.

- **Role incongruence**: a function of role status and role expectancy is updated as roles change.

- **Role strain**: a function of role overload and role incongruence is updated as roles change.

- **Role socialisation**: a function of long-run desire to drink and role status is updated as the long run desire to drink and roles change.

- **Drinking opportunity**: a function role status is updated as roles change.

4.6 The action mechanisms determine agent drinking based on three variables: role strain, role socialisation, and drinking opportunity. Lastly, the transformational mechanisms update the societal transition rates based on the drinking of the agents, e.g. agents that drink more are more likely to transition out of employment. An implementation of the social roles model was reported in Bai et al. (2019).

Design of a combined norms and social roles model

4.7 The MBSSM architecture of norms theory was extended to add roles theory mechanisms by going through the OOAD process. We added classes related to social roles theory by extending base classes. Figure 6 shows the class diagram of the final design with the new classes in orange (compared to Figure 4). Four classes were added: `RoleTheory`, `RoleEntity`, `CombinedModel`, `WeightedSumTheoryMediator`. Two classes from the norms model are no longer used: `NormModel` and `OneTheoryMediator`, and are greyed out in Figure 6.

- The `RoleTheory` class has role-related attributes: role status, role overload, role incongruence, drinking opportunity, role strain, and role socialisation. `RoleTheory` also implements operations for situational and action mechanisms.
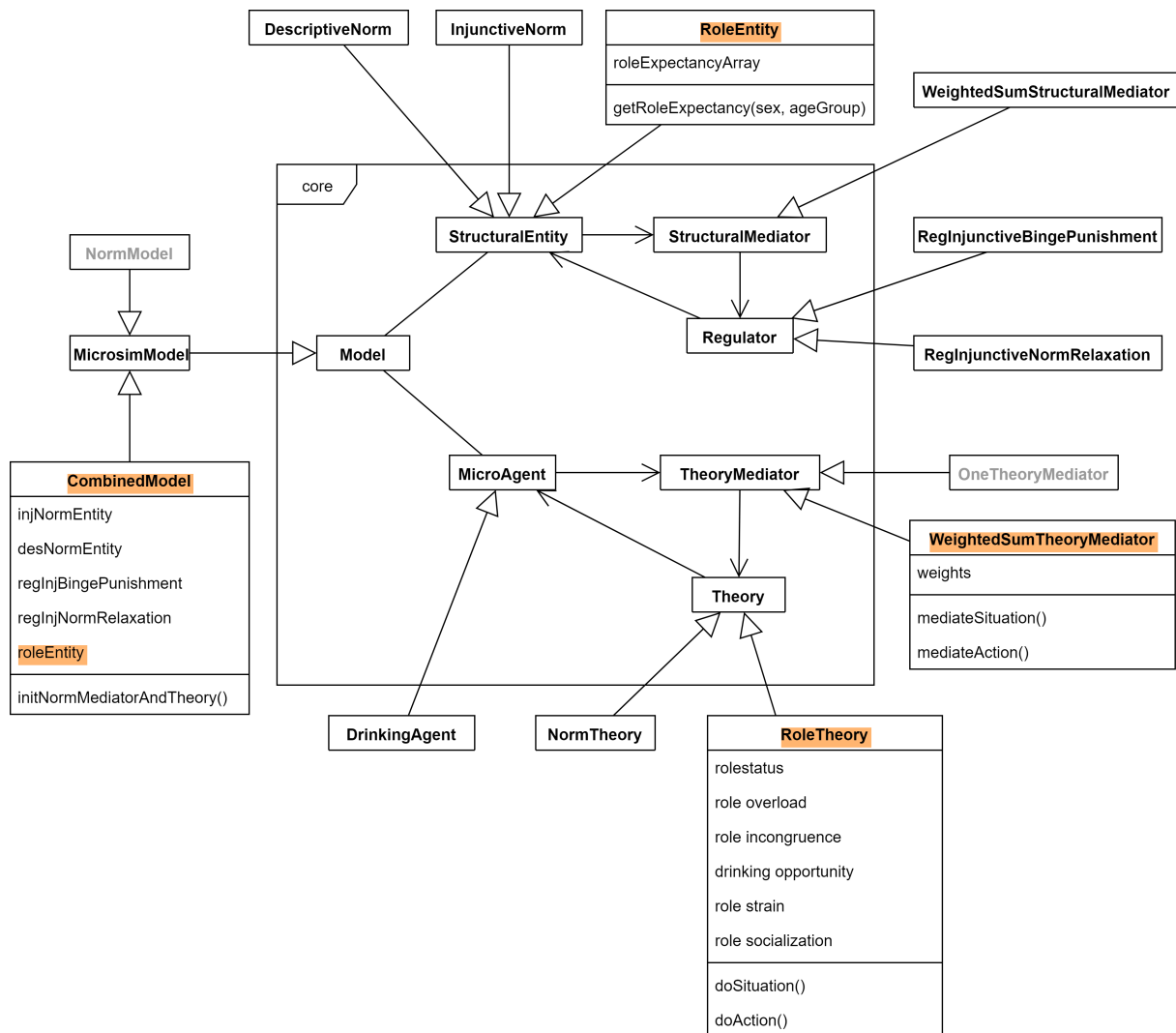
Figure 6: Class diagram for a hybrid model of norms and roles theory. Note: The classes in grey are no longer used. The classes highlighted in orange are new elements in the hybrid model, compared to the single-theory norms model (Figure 4).

- The `RoleEntity` class represents societal expectation in term of role status.

- The `CombinedModel` class was added to assign two theories when initialising an agent (which is different from `NormModel` that assigns one theory). The `CombinedModel` class also manages the societal transition rates.

- The `WeightedSumTheoryMediator` class was introduced to mediate the different drinking decisions from both theories into a final drinking decision of an agent. There can be many different approaches to combine theories and each approach can be implemented as a class so that the system has a collection of different `TheoryMediator` objects.

4.8 In summary, the modelling workflow when adding a theory using the MBSSM architecture is:

- Extend from the `Theory` class to create classes related to the new theory;

- Decide how to combine different theories and write a class extended from `TheoryMediator` to link agent decision with different theories;

- Extend from the `StructuralEntity` class to include required theory-specific structural entities as well as Regulator classes if needed;

- Adapt the `Model` class to initialise agents, theories, mediator to link agents and theories, as well as related structural entities and regulators.

Discussion

Summary of key novelty and findings

5.1 To the best of our knowledge, this is the first paper to implement a software architecture for mechanism-based social systems modelling. It encourages conceptualising a theory using three types of mechanisms (situational, action, transformational mechanisms) to describe the interaction between macro and micro levels in order to construct an operational mechanism-based theory model. MBSSM, by providing a framework that was developed from those aspects of sociological meta-models that are shared across all models, establishes a theory-supported basis for modellers to build social simulations that are not ad hoc, but rooted in sociological discourse. Modellers are free to model according to a specific epistemological stance at the model conceptualisation stage. Once operationalised in a computer program, these social simulation models could be used to investigate the explanatory value of the theory and even, if considered appropriate, the predictions for the future of the social system based on the theory.

5.2 Additionally, the MBSSM architecture provides a formal approach for multi-theory modelling. Separate models developed from different social theories can now be projected into a general micro-macro scheme. By decomposing the elements and processes of a social theory, a library of theory components can be accumulated. The library can then be exploited to developed collections of mechanisms (and therefore integrations of theories) that interact with each other. The multi-theory approach enriches the chain of causal mechanisms to better explain the social system than any single theory could.

5.3 Finally, the MBSSM provides a strong foundation for enabling researchers to perform both parameter and structural calibration so that the model can potentially satisfy both empirical and theoretical plausibility. Hence, MBSSM has a strong enabling link to the ongoing programme of research on the calibration of agent-based models. Thiele et al. (2014) provides a good summary and examples of different methods of parameter calibration and sensitivity analysis. For parameter calibration, examples are given for sampling techniques (simple random sampling, Latin hypercube sampling) to optimisation methods (gradient and quasi-Newton methods, simulated annealing, evolutionary algorithms) to Bayesian methods. We note that most research on calibration relates to parameter calibration. The structural aspect of the model is usually carefully designed but then fixed (and therefore often implicitly assumed 'perfect') before undertaking parameter calibration. What the current research landscape lacks is better consideration of structural calibration and theoretical standards, which are arguably the requirements for analytical models where parameters reflect theoretical plausibility. MBSSM offers a common template for representing, communicating, integrating and re-using alternative theory-grounded structural assumptions. We believe this has great potential for structural calibration.

Strength and limitations

5.4 The first advantage of the MBSSM is that the software engineering principle of separation of concerns is held during its application. The classes in the core package enforce the general micro-macro scheme for mechanism-based social systems and provide a skeleton or template for implementation of different models. The core classes usually stay the same during the application of the MBSSM architecture. The theory-specific classes are designed separately; thus there is a separation between different theory-specific functionalities. The software architecture is robust because a modeller can add a theory without affecting the functionality of other theories; if there are many theories implemented, one can choose which theory classes to use; different theory mediators for different combination approaches can be implemented and added to a collection. Another strength of the MBSSM architecture is that it is independent of any particular design process and software tool. That means the architecture can be used by different practitioners and models can be implemented in any software toolkit. Lastly, since UML diagrams are used during development, these diagrams become design documentation as a by-product of the development process and can be used for better communication between modellers and researchers from other disciplines.

5.5 The MBSSM software architecture has the following limitations. Firstly, since MBSSM architecture is based on the general micro-macro scheme, theories must be represented using macro-micro-macro middle-range meta-theory to be incorporated into the model. Secondly, MBSSM architecture does not explicitly support the dynamic formation of macro-level entities (the system cannot create novel social structures during the simulation run). The architecture has a static ontology that comes from the modeller it is not an AI that can create its own ontologies. So, the macro *concepts* of, for example, 'norm', 'identity', 'network', or 'role' are pre-defined

by the modeller. However, the *content* of these macro concepts can be dynamic, and hence we are able to calibrate model structure within the static ontological framework. For example, types of identity (i.e. groupings of agents based on similarity in terms of properties and/or behaviour) can be generated dynamically as part of the model. In general, the macro entities have states that change over the course of the simulation. Thirdly, the modellers need to acquire basic knowledge about UML diagrams and familiarise themselves with this architecture. Fourthly, since the MBSSM architecture was designed under the object-oriented programming paradigm, the model implementation needs to use the same paradigm. This means the architecture is not straightforwardly compatible with other paradigms like purely functional programming language (such as Haskell (Thaler et al. 2018)) or special-purpose multi-paradigm languages (e.g. NetLogo Wilnesky 1999 is implemented in Scala and Java (OOP) but the coding by modellers is performed in the bespoke NetLogo language, a variant of Logo, which allows restricted control of active objects and decomposing the program into a collection of variables, procedures, and potential data structures). Finally, the architecture design does not explicitly support changing model structure at run time. When changing the structure of low-level primitives such as equations or statements the necessary components of the model could be recompiled before the evaluation run. As another option, modellers could design a grammar for configuration of model behaviour so that the model does not have to be recompiled but can read in these configurations and change its behaviour accordingly.

Implications for different stakeholders: Researchers, modellers, decision makers

5.6 A modelling process can involve many stakeholders with different interests and expectations. For certain, analytical sociologists focus on whether models can provide explanations: plausible mechanisms that explain why phenomena occur. On the other hand, policy makers consider the models useful if they support predictions and policy experimentation. They may not value mechanism-based explanations much, and might even consider these mechanisms too complicated (Boero & Squazzoni 2005). Therefore, it is important to clearly define modelling purposes since they affect the other activities of developing models, experimenting, and presenting results (Edmonds et al. 2019). This section presents implications to different stakeholders when applying the MBSSM architecture to the modelling lifecycle of mechanism-based social systems.

5.7 Social systems researchers can use our architecture to develop models to systematically explore the "cogs and wheels of the internal machinery" (Elster 1989) of social phenomena and gain a better understanding of the social world. Since different mechanisms can be embedded in models and these models can be experimented with, researchers can use the models for theory exploration and development. This would be beneficial to any disciplines working with social systems, including analytical sociology, psychology, ecology, and economics.

5.8 Modellers working with social systems can use the MBSSM architecture as a starting point to accelerate the development process. Even when the modellers only consider a social theory, the resultant model can also be easily extended as needed to add mechanisms from other theories. We provided an implementation of the architecture in RepastHPC; however, the architecture can be implemented in any software or toolkit.

5.9 Lastly, decision makers need to be aware of the possibility and capability of multi-theory explanation since a theory can fill in the explanatory gaps that another theory is missing. They can encourage and propose using multiple theories as long as the theories are formalised in the same scheme and the model is designed, developed, tested, and validated in a transparent process. The MBSSM architecture supports such rigorous development of mechanism-based social systems.

Implications for further research needed and opportunities this provides to develop the field

Approaches to combine mechanism-based social theories

5.10 When building a model that contains several theories or social mechanisms, many difficult choices must be made about how to combine theories and mediate social mechanisms. This is not only an issue in multi-theory models but also in single-theory models. Even within a single theory, different mechanisms can pull in different directions, e.g. injunctive norm can be restricted or relaxed by two different transformational mechanisms. In the following, we discuss combination approaches that exploit the mediator in the MBSSM architecture. We propose several ways to combine mechanisms.

5.11 A way of combining theory is the weighted mean of decisions that we demonstrated in the two case studies. Theory classes usually share a common ontology for the connection with the decision of the `DrinkingAgent` class. In this case study of alcohol modelling, it is the drinking decision. In the simplest case, the drinking decision can be a number of drinks. Another way to model the decision is as a series of drinking dispositions: the probability of having a drink based on the number of previous drinks. The weighted mean can be performed to either the number of drinks or drinking dispositions. The drinking disposition is a series of probabilities, and there can be different weights for each probability. For example, if we assume that norms greatly influence the decision to have the first drink while roles will be more important from the second drink onward, the modeller can set the weights of norms and roles for the first drink to 0.9 and 0.1 respectively and set the weights to 0.5 and 0.5 from the second drinks onward. Such weightings can then be subject to calibration.

5.12 Theory combination can also be performed at a lower level with a rule-based or heuristic-based approach. At the low level, the `Mediator` class can work with attributes and operations exposed by the `Theory` classes and perform complex synthesis as according to a series of rules. For theory combination at a higher level of mechanisms, modellers can include or exclude different operations inside mechanisms or rearrange their order to study the effects on the mechanism. Combining models this way can result in a large number of different model structures comprised of the same basic elements, which leads to the problem: which structure should we choose to provide a useful model. In the next section, we discuss structural calibration as a solution to this problem.

Structural calibration

5.13 Let us use an example of an agent-based simulation study where multiple theories are implemented. From the previous discussion, the software architecture shows its capability to support multiple theories and flexible structural changes at different levels of abstraction (behaviour of both micro-level agents and macro-level structural entities). Alternative structures can be tried out to find a 'better' model: either replicating an observed emergence or better fitting to the empirical data. As introduced in Section 2.19, this process is called structural calibration.

5.14 In structural calibration, modellers define *building blocks* in the model and the structural calibration process identifies the most promising combinations of those building blocks. At the highest level of abstraction of the social systems, we can change the order of three mechanisms between different theories: the `TheoryMediator` can manipulate `doSituation()` and `doAction()` operations in the Theory class while the Model controls the `doTransformation()` operation of `StructuralEntity`. At the lowest level of abstraction, the equations within the mechanisms can be altered (Vu et al. 2019a). For example, the drinking equation in the norms model is influenced by four attributes of each person: perceived injunctive norm, perceived descriptive norm, desire to drink, and autonomy. The drinking equation can combine these four attributes in a different way or we can even introduce more attributes to this function. Such investigation could prove important a recent study (Muelder & Filatova 2018) demonstrated how different interpretations of the Theory of Planned Behaviour led to qualitatively different outcomes from a model. After the process of structural calibration, modellers can investigate the resultant structures to validate theories, study the relationships of different processes in a theory, understand which components in the theories are more important, and even propose a new theory (if a new structure emerges after such calibration).

5.15 So how should we perform structural calibration? With models of complex systems, brute force search through all structures is usually not feasible. Machine learning techniques can provide solutions for the structural calibration of agent-based models. However, since our software architecture is designed for mechanism-based social systems, the resulting structures need to be interpretable and explainable so that domain experts can

use them for theory validation and exploration. Therefore, machine learning techniques that result in a black box model will not be suitable. For example, a deep neural network can approximate a function with an interlaced multi-layer web of connections but this complex structure is almost impossible to interpret. A more interpretable approach is reasonable-size regression equations, decision trees, or a syntax tree. For instance, a tree-based representation is popular in genetic programming techniques because it is expressive and can be used for statements, rules, and equations. It is also feasible for domain experts to study a tree-based representation with reasonable depth. In summary, the representation of the structures has to be expressive and explicit, and machine learning techniques can be used to automatically explore structures. In Vu et al. (2019a), genetic programming was used with MBSSM; eight alternative model structures were identified, representing different trade-offs between model complexity and empirical plausibility: In Vu et al. (2020), grammar-based genetic programming was used to structurally calibrate the social roles model, with the results scrutinised by a domain expert. In this latter case, the approach was successful in identifying three competing explanations for the observed societal patterns of alcohol use, using novel integrations of role theory not previously considered by the human modeller.

The interaction between structural and parameter calibration

5.16   There is an interaction between structural calibration and parameter calibration. In parameter calibration, the parameters are adjusted to provide the outputs with better fit to the data. On the other hand, structural calibration focuses on the structure of model i.e. behaviour rules and mechanisms. A different structure changes the causal chain of mechanisms and can require a different parameter set to fit empirical data. Therefore, during structural calibration, evaluating a new structure should require a new parameter calibration. Doing parameter calibration within structural calibration is computationally expensive. Depending on the parameter space and structure space, this may not feasible. One possible solution is developing a surrogate model to estimate the fitness for the new structure, where the use of black-box models like neural networks becomes more appropriate. Future research on solving this problem is needed, but a small literature is emerging related topics (Hildebrandt & Branke 2015; Sinha et al. 2018).

Conclusions

5.17   In this work, the need for a unified software architecture for modelling mechanism-based social systems was identified and the rationale behind the design was explained in detail. Software engineering principles were incorporated into the design to ensure usability and extensibility. Then to demonstrate the architecture, an agent-based simulation study was described, using two social theories: the theory of normative behaviour, and role theory. We also discuss how this software architecture supports the combination of theories via the mediator design pattern and discuss different approaches to combine mechanisms from multiple theories. Lastly, because the MBSSM architecture enforces the mechanism-based modelling and explanation via the general micro-macro scheme, it provides an environment to open the black box of a social system and study the internal mechanisms.

5.18   Future planned work includes applying the MBSSM architecture to different theories as well as a combination of theories with the exploitation of the mediator. With a multi-theory model, it is also interesting to design and perform a structural calibration process for theory exploration and development. Structural calibration presents many new research challenges: choosing methodology for structural calibration, interfacing between the structural calibration and the model, combining structural calibration with parameter calibration, and interactive engagement with domain experts during the calibration process.

Acknowledgements

References

Ajzen, I. (1991). The theory of planned behavior. *Organizational Behavior and Human Decision Processes*, *50*(2), 179–211

Archer, M. S. (1995). *Realist Social Theory: The Morphogenetic Approach*. Cambridge: Cambridge university press

Bai, H., Brennan, A., Broomhead, T., Meier, P. S., Nielsen, A., Probst, C., Tuong, V. & Purshouse, R. C. (2019). Modeling alcohol use behavior at population scale based on social role theory: An exploratory agent-based model. In *HEDS Discussion Paper 17/04*. Sheffield: ScHARR, University of Sheffield

Beaumont, M. A. (2010). Approximate Bayesian computation in evolution and ecology. *Annual Review of Ecology, Evolution, and Systematics*, *41*(1), 379–406

Becker, G. S. & Murphy, K. M. (1988). A theory of rational addiction. *Journal of Political Economy*, *96*(4), 675–700

Bersini, H. (2011). UML for ABM. *Journal of Artificial Societies and Social Simulation*, *15*(1), 9

Boero, R. & Squazzoni, F. (2005). Does empirical embeddedness matter? Methodological issues on agent-based models for analytical social science. *Journal of Artificial Societies and Social Simulation*, *8*(4), 6

Bourgais, M., Taillandier, P., Vercouter, L. & Adam, C. (2018). Emotion modeling in social simulation: A survey. *Journal of Artificial Societies and Social Simulation*, *21*(2), 5

Christakis, N. A. & Fowler, J. H. (2013). Social contagion theory: Examining dynamic social networks and human behavior. *Statistics in Medicine*, *32*(4), 556–577

Cialdini, R. B., Kallgren, C. A. & Reno, R. R. (1991). A focus theory of normative conduct: A theoretical refinement and reevaluation of the role of norms in human behavior. In M. P. Zanna (Ed.), *Advances in Experimental Social Psychology*, vol. 24, (pp. 201–234). New York, NY: Academic Press

Coleman, J. S. (1986). Social Theory, Social Research, and a Theory of Action. *American Journal of Sociology*, *91*(6), 1309–1335

Collier, N. & North, M. (2013). Parallel agent-based simulation with Repast for High Performance Computing. *Simulation*, *89*(10), 1215–1235

Collins, A., Petty, M., Vernon-Bido, D. & Sherfey, S. (2015). A Call to Arms: Standards for Agent-Based Modeling and Simulation. *Journal of Artificial Societies and Social Simulation*, *18*(3), 12

Danermark, B., Ekström, M., Jakobsen, L. & Ch. Karlsson, J. (2002). *Explaining Society: Critical Realism in the Social Sciences*. London: Routledge

Edmonds, B., Le Page, C., Bithell, M., Chattoe-Brown, E., Grimm, V., Meyer, R., Montañola-Sales, C., Ormerod, P., Root, H. & Squazzoni, F. (2019). Different Modelling Purposes. *Journal of Artificial Societies and Social Simulation*, *22*(3), 6

Elster, J. (1989). *Nuts and Bolts for the Social Sciences*. Cambridge: Cambridge University Press

Epstein, J. M. (1999). Agent-based computational models and generative social science. *Complexity*, *4*(5), 41–60

Epstein, J. M. (2006). *Generative Social Science: Studies in Agent-Based Computational Modeling*. Princeton, NJ: Princeton University Press

Fowler, M., Kobryn, C. & Scott, K. (2004). *UML distilled: a brief guide to the standard object modeling language*. Addison-Wesley Professional

Gamma, E. (1995). *Design Patterns: Elements of Reusable Object-Oriented Software*. Boston, MA: Addison-Wesley

Gilbert, N. & Troitzsch, K. (2005). *Simulation for the Social Scientist*. New York, NY: McGraw-Hill

Grazzini, J., Richiardi, M. G. & Tsionas, M. (2017). Bayesian estimation of agent-based models. *Journal of Economic Dynamics and Control*, *77*, 26–47

Grimm, V., Revilla, E., Berger, U., Jeltsch, F., Mooij, W. M., Railsback, S. F., Thulke, H.-H., Weiner, J., Wiegand, T. & DeAngelis, D. L. (2005). Pattern-oriented modeling of agent-based complex systems: Lessons from ecology. *Science*, *310*(5750), 987–991

Gunaratne, C. & Garibay, I. (2017). Alternate social theory discovery using genetic programming: Towards better understanding the artificial anasazi. In *GECCO 2017 - Proceedings of the 2017 Genetic and Evolutionary Computation Conference*, (pp. 115–122). Berlin: ACM Press

Hedström, P. & Bearman, P. (2009). *The Oxford Handbook of Analytical Sociology*. Oxford: Oxford University Press

Hedström, P. & Swedberg, R. (1996). Social mechanisms. *Acta Sociologica*, *39*(3), 281–308

Hildebrandt, T. & Branke, J. (2015). On using surrogates with genetic programming. *Evolutionary Computation*, *23*(3), 343–367

Jayne, M., Valentine, G. & Holloway, S. L. (2008). Fluid boundaries — British binge drinking and European civility: Alcohol and the production and consumption of public space. *Space and Polity*, *12*(1), 81–100

Jepperson, R. & Meyer, J. W. (2011). Multiple levels of analysis and the limitations of methodological individualisms. *Sociological Theory*, *29*(1), 54–73

Knibbe, R., Drop, M. & Muytjens, A. (1987). Correlates of stages in the progression from everyday drinking to problem drinking. *Social Science and Medicine*, *24*(5), 463–473

Kuntsche, S., Knibbe, R. A. & Gmel, G. (2009). Social roles and alcohol consumption: A study of 10 industrialised countries. *Social Science and Medicine*, *68*(7), 1263–1270

Le, V. M., Adam, C., Canal, R., Gaudou, B., Vinh, H. T. & Taillandier, P. (2012). Simulation of the emotion dynamics in a group of agents in an evacuation situation. In N. Desai, A. Liu & M. Winikoff (Eds.), *Artificial Intelligence: Theories, Models and Applications*, (pp. 604–619). Berlin/Heidelberg: Springer

Lenormand, M., Jabot, F. & Deffuant, G. (2013). Adaptive approximate Bayesian computation for complex models. *Computational Statistics*, *28*(6), 2777–2796

León-Medina, F. J. (2017). Analytical sociology and agent-based modeling: Is generative sufficiency sufficient? *Sociological Theory*, *35*(3), 157–178

Lomax, N. & Norman, P. (2016). Estimating population attribute values in a table: "Get me started in" iterative proportional fitting. *The Professional Geographer*, *68*(3), 451–461

Manzo, G. (2007). The progress and "urgency" of modeling in sociology. The concept of the "generative model" and its implementation. *L'année sociologique*, *57*(1), 13–61

Manzo, G. (2014). *Analytical Sociology: Actions and Networks*. Chichester: Wiley

Marchionni, C. & Ylikoski, P. (2013). Generative explanation and individualism in agent-based simulation. *Philosophy of the Social Sciences*, *43*(3), 323–340

Mayntz, R. (2004). Mechanisms in the analysis of social macro-phenomena. *Philosophy of the Social Sciences*, *34*(2), 237–259

Miller, K. D. (2015). Agent-based modeling and organization studies: A critical realist perspective. *Organization Studies*, *36*(2), 175–196

Muelder, H. & Filatova, T. (2018). 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*, *21*(4), 5

Neighbors, C., O'Connor, R. M., Lewis, M. A., Chawla, N., Lee, C. M. & Fossos, N. (2008). The relative impact of injunctive norms on college student drinking: The role of reference group. *Psychology of Addictive Behaviors*, *22*(4), 576–581

Probst, C., Greenfield, T. K., Nielsen, A., Shuper, P. A., Purshouse, R. C. & Rehm, J. (2020a). Convergence of gender-specific drinking norms and patterns of alcohol use 1979-2016 in the US. Manuscript submitted for publication

Probst, C., Vu, T. M., Epstein, J. M., Nielsen, A. E., Buckley, C., Brennan, A., Rehm, J. & Purshouse, R. C. (2020b). The normative underpinnings of population-level alcohol use: An individual-level simulation model. *Health Education & Behavior*, *47*(2), 224–234

Purshouse, R. C., Ally, A. K., Brennan, A., Moyo, D. & Norman, P. (2014). Evolutionary parameter estimation for a theory of planned behaviour microsimulation of alcohol consumption dynamics in an English birth cohort 2003 to 2010. In *Proceedings of the 2014 Conference on Genetic and Evolutionary Computation - GECCO '14*, (pp. 1159–1166)

Ramström, G. (2018). Coleman's boat revisited: Causal sequences and the micro-macro link. *Sociological Theory*, *36*(4), 368–391

Rimal, R. N. & Real, K. (2005). How behaviors are influenced by perceived norms: A test of the theory of normative social behavior. *Communication Research*, *32*(3), 389–414

Room, R., Osterberg, E., Ramstedt, M. & Rehm, J. (2009). Explaining change and stasis in alcohol consumption. *Addiction Research & Theory*, *17*(6), 562–576

Sawyer, R. K. (2005). *Social emergence: Societies as complex systems*. Cambridge University Press

Schelling, T. C. (1971). Dynamic models of segregation. *The Journal of Mathematical Sociology*, *1*(2), 143–186

Scott, S., Shucksmith, J., Baker, R. & Kaner, E. (2017). 'hidden habitus': A qualitative study of socio-ecological influences on drinking practices and social identity in mid-adolescence. *International Journal of Environmental Research and Public Health*, *14*(6), 611

Sinha, A., Malo, P. & Deb, K. (2018). A review on bilevel optimization: From classical to evolutionary approaches and applications. *IEEE Transactions on Evolutionary Computation*, *22*(2), 276–295

Steglich, C., Snijders, T. A. & Pearson, M. (2010). Dynamic networks and behavior: Separating selection from influence. *Sociological Methodology*, *40*(1), 329–393

Stonedahl, F. & Wilensky, U. (2011). Finding forms of flocking: Evolutionary search in ABM parameter-spaces. In T. Bosse, A. Geller & C. M. Jonker (Eds.), *Multi-Agent-Based Simulation XI*, (pp. 61–75). Berlin/Heidelberg: Springer

Thaler, J., Altenkirch, T. & Siebers, P.-O. (2018). Pure functional epidemics: An agent-based approach. In *Proceedings of the 30th Symposium on Implementation and Application of Functional Languages*, IFL 2018, (pp. 1–12). Lowell, MA, USA: Association for Computing Machinery

Thiele, J., Kurth, W. & Grimm, V. (2014). Facilitating parameter estimation and sensitivity analysis of agent-based models: A cookbook using NetLogo and 'R'. *Journal of Artificial Societies and Social Simulation*, *17*(3), 11

Tilly, C. (2001). Mechanisms in political processes. *Annual Review of Political Science*, *4*(1), 21–41

van der Vaart, E., Beaumont, M. A., Johnston, A. S. A. & Sibly, R. M. (2015). Calibration and evaluation of individual-based models using Approximate Bayesian Computation. *Ecological Modelling*, *312*, 182–190

Verhagen, H. (2001). Simulation of the learning of norms. *Social Science Computer Review*, *19*(3), 296–306

Vu, T. M., Buckley, C., Bai, H., Nielsen, A., Probst, C., Brennan, A., Shuper, P., Strong, M. & Purshouse, R. C. (2020). Multiobjective genetic programming can improve the explanatory capabilities of mechanism-based models of social systems. *Complexity*, *2020*, 8923197

Vu, T. M., Probst, C., Epstein, J. M., Brennan, A., Strong, M. & Purshouse, R. C. (2019a). Toward inverse generative social science using multi-objective genetic programming. In *GECCO 2019 - Proceedings of the 2019 Genetic and Evolutionary Computation Conference*, GECCO '19, (pp. 1356–1363). Prague, Czech Republic: Association for Computing Machinery

Vu, T. M., Wagner, C. & Siebers, P.-O. (2019b). ABOOMS: Overcoming the hurdles of continuous-time public goods games with a simulation-based approach. *Journal of Artificial Societies and Social Simulation*, *22*(2), 7

Wilnesky, U. (1999). *NetLogo*. Center of Connected Learning and Computer-Based Modeling, Norwestern University, Evanston, IL

Wilsnack, R. W. & Cheloha, R. (1987). Women's roles and problem drinking across the lifespan. *Social Problems*, *34*(3), 231–248

Yamaguchi, K. & Kandel, D. B. (1985). On the resolution of role incompatibility: A life event history analysis of family roles and marijuana use. *American Journal of Sociology*, *90*(6), 1284–1325

Ylikoski, P. (2016). *Thinking with the Coleman Boat*. Linköping: Linköping University Electronic Press

Zoumpoulaki, A., Avradinis, N. & Vosinakis, S. (2010). A multi-agent simulation framework for emergency evacuations incorporating personality and emotions. In S. Konstantopoulos, S. Perantonis, V. Karkaletsis, C. D. Spyropoulos & G. Vouros (Eds.), *Artificial Intelligence: Theories, Models and Applications*, (pp. 423–428). Berlin/Heidelberg: Springer