Chapter 9
Verifying and Validating Simulations

Nuno David, Nuno Fachada, and Agostinho C. Rosa

Abstract Verification and validation are two important aspects of model building. Verification and validation compare models with observations and descriptions of the problem modelled, which may include other models that have been verified and validated to some level. However, the use of simulation for modelling social complexity is very diverse. Often, verification and validation do not refer to an explicit stage in the simulation development process, but to the modelling process itself, according to good practices and in a way that grants credibility to using the simulation for a specific purpose. One cannot consider verification and validation without considering the purpose of the simulation. This chapter deals with a comprehensive outline of methodological perspectives and practical uses of verification and validation. The problem of evaluating simulations is addressed in four main topics: (1) the meaning of the terms verification and validation in the context of simulating social complexity; (2) types of validation, as well as techniques for validating simulations; (3) model replication and comparison as cornerstones of verification and validation; and (4) the relationship of various validation types and techniques with different modelling strategies.

Why Read This Chapter?

To help you decide how to check your simulation—both against its antecedent conceptual models (verification) and external standards such as data or other simulations (validation)—and in this way help you to establish the credibility of your simulation. In order to do this the chapter will point out the nature of these processes, including the variety of ways in which people seek to achieve them.

The original version of this chapter was revised. An erratum to this chapter can be found at https://doi.org/10.1007/978-3-319-66948-9_30

N. David (✉)
DINÂMIA'CET - ISCTE-IUL - Centre for Socioeconomic and Territorial Studies, ISCTE-IUL Instituto Universitário de Lisboa, Av. das Forças Armadas, 1649-026 Lisboa, Portugal
e-mail: nuno.david@iscte.pt

N. Fachada • A.C. Rosa
Institute for Systems and Robotics (ISR/IST), LARSyS, Instituto Superior Técnico, Av. Rovisco Pais, 1, 1049-001 Lisboa, Portugal
e-mail: nfachada@laseeb.org; acrosa@laseeb.org

B. Edmonds, R. Meyer (eds.), *Simulating Social Complexity*,
Understanding Complex Systems, https://doi.org/10.1007/978-3-319-66948-9_9

9.1 Introduction

The terms verification and validation (V&V) are commonly used in science but their meaning may be controversial in the natural and the social sciences. Putting aside the epistemological underpinnings of the terms, in simulation the distinction of meaning has a mere pragmatic nature inherited from computer science and software engineering. Often, *verification* is used in the context of evaluating the computational implementation of a model in terms of the researchers' intentions. In turn, *validation* typically refers to an evaluation of the credibility of the model as a representation of the subject modelled.

In disciplines that make use of computational models, the role of V&V is related to the need of evaluating models along the simulation development process. Basically, the very idea of V&V is comparing models with observations and descriptions of the problem modelled. This may include other models that have been verified and validated to some level, or even the implementation of replications in order to verify and validate models in more depth.

This chapter introduces a methodological perspective on V&V and describes different strategies and techniques to validate models of social complexity. Some aspects of what can be called either verification or validation are also discussed, namely comparison between models and model replication, whereon verification and validation are superimposed or indistinguishable. These are important but frequently neglected methods of promoting V&V, particularly since social simulation models can be very sensitive to implementation details (making them hard to verify), and data from social systems can be difficult or even impossible to collect (making the respective models hard to validate).

The use of simulation for modelling social complexity is very diverse. Often, V&V do not refer to an explicit stage in the simulation development process, but to the modelling process itself according to good practices and in a way that grants credibility to using the simulation for a specific purpose. Normally, the purpose is dependent on different strategies and dimensions, along which simulations can be characterised, with reference to different kinds of claims intended by the modeller, such as theoretical claims, empirical claims or simply subjunctive theoretical claims. The term subjunctive is used when very abstract simulations are used for thinking about scenarios in possible worlds, such as describing "what *would* happen if something were the case." There cannot be V&V without considering the purpose of the simulation.

In the next section of the chapter, we will deal with the meaning of the terms V&V in the context of the simulation development process. In Sect. 9.3, methods and techniques commonly associated with validation are described. The comparison and replication of simulation models as an essential aspect of V&V is discussed in Sect. 9.4. The chapter closes with Sect. 9.5, where the relationship of validation with different modelling strategies is described.

9.2 The Simulation Development Process

Several chains of intermediate models are developed before obtaining a satisfactory verified and validated model. What does it mean to verify and validate a model in social simulation? Is there a fundamental difference between verifying and validating models? The purpose of this section is to define the role of V&V within the scope of the simulation development process.

The most common definitions of V&V are imported from computer science, as well as from technical and numerical simulation,[1] having intended distinct—although epistemologically overlapping—meanings. The reason for distinguishing between the terms derives from the practice of determining the suitability of certain models for representing two distinct subjects of inquiry. This is represented in Fig. 9.1, in which V&V are related to a simplified model development process. Two conceptual models mediate between two subjects of inquiry. The latter are (1) the target theory or phenomenon and (2) the executable computational model. The conceptual model on the right, designated here as the *pre-computational model*, is basically a representation in the minds and writing of the researchers, which presumably represents the target. This model must be implemented as an *executable computational model*, by going through a number of intermediate models such as formal specification or textual programs written in high-level programming languages.

The analysis of the executable model gives rise to one or more conceptual models on the left, here designated as *post-computational* models. They are constructed based on the output of the computational model, often with the aid of statistical

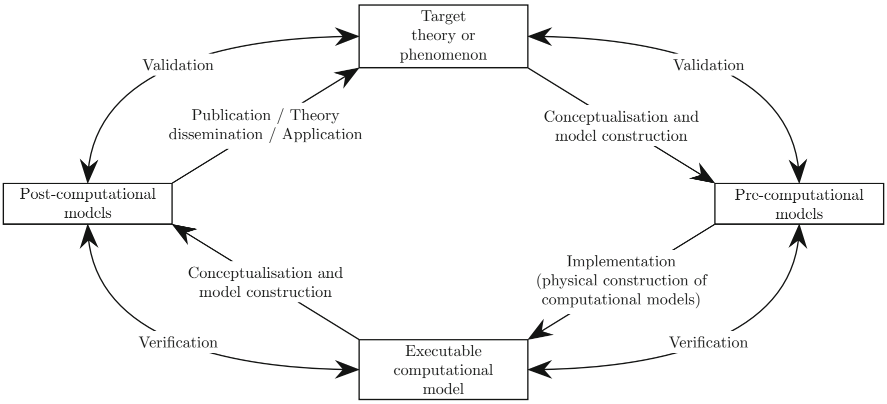

Fig. 9.1 Verification and validation related to the model development process (David 2009)

[1]Numerical simulation refers to simulation for finding solutions to mathematical models, normally for cases in which mathematics does not provide analytical solutions. Technical simulation stands for simulation with numerical models in computational sciences and engineering.

packages, graphing and visualisation. The whole construction process results in categories of description that may not have been used for describing the pre-computational model. This is the so-called idea of *emergence*, when interactions among model components specified through pre-computational models at some level of description give rise to different categories of model descriptions identified in the executable model at macro levels of observation, expressed through post-computational models.

As an example consider the culture dissemination model of Axelrod (1997b) which has a goal of analysing the phenomena of social influence. At a micro-level of description, a pre-computational model defines: (a) the concept of *actors* distributed on a grid; (b) the concept of *culture* of each actor, specified as a set of five features; and (c) the *interaction mechanisms* specified with a bit-flipping schema, in which the probability of interaction between two actors is set proportionately to the similarity between two cultures. The executable model is then explored and other categories of descriptions resulting from the interaction of individual cultures may be defined. These are associated with macro properties of interest and conditions in which they form, such as the concepts of *regions* and *zones* on the grid. A great deal of the simulation proposed by Axelrod concerns investigating properties of *regions* and *zones* in the executable model, giving rise to a proposed conceptual, post-computational model, which expresses traits such as the relation between the size of a *region* formed and the number of features per *individual culture*. These concepts are interpreted in relation to the target social phenomena of social influence.

We will now situate the role of V&V in the modelling process of social simulation.

9.2.1 What Does It Mean to Verify a Computational Model?

Computational model *verification* is defined as checking the adequacy among conceptual models and computational models (see also Chap. 7 in this volume, Galán et al. 2017). Consider the lower quadrants of Fig. 9.1. They are concerned with ensuring that the pre-computational model has been implemented *adequately* as an executable computational model, according to the researcher's intentions in the parameter range considered, and also that the post-computational model *adequately* represents the executable model in the parameter range considered.[2] In short, the three models must correspond to each other adequately, relative to the same target they are meant to represent.

At this point you might question the meaning of *adequately*. A minimal definition could be the following: adequateness means that the inputs, outputs and the mechanisms post-computationally modelled from the executable computational model are consistent with the ones specified through the pre-computational models,

[2] Verification in the left quadrant of Fig. 9.1 is sometimes known as "internal validation."

in accordance with the researcher's intentions. However, the outcomes of computer programs in social simulation are often unintended or not known a priori and thus the verification process requires more than checking that the executable model does what it was planned to do. The goal of the whole exercise is to assess logical inferences within, as well as between, the pre- and the post-computational models. This requires assessing whether the post-computational model—while expressing emergent concepts that the pre-computational model may not have been intended to express—is consistent with the latter. From a methodological point of view this is a complicated question, but from a practical perspective one might operationally define the verification problem with the following procedures:

(a) For some pre-computational model definable as a set of input/output pairs *in a specified parameter range*, the corresponding executable model is *verified for the range considered* if the corresponding post-computational model expresses the same set of inputs/outputs for the range considered.
(b) For some pre-computational model defined according to the researcher and/or stakeholders' intentions *in a specified parameter range*, the corresponding executable model is *verified for the range considered* if the corresponding post-computational model meets the researchers and/or stakeholders' expectations for the range considered.

Note that both procedures limit the verification problem to a clearly defined parameter range. The first option is appropriate when quantitative data is available from the target with which to test the executable model. This is normally not the case, leaving the second option as the suitable path for the verification process. This is possible since the aim is to assess the appropriateness of the relations that may be established between micro-levels of description specified in the pre-computational model and macro-levels of description expressed through post-computational models, usually amenable to evaluation by researchers and stakeholders.

In any case, the verifiability of a simulation is influenced by the process used to develop that simulation. The tools used to implement the executable computational model are a major factor affecting verification (Sargent 2013). The use of high-level simulation packages has the potential to simplify verification, since the majority of common model building blocks are provided, and these are typically already verified. Arguably, this is even more so in the case of open source toolkits, such as NetLogo (Wilensky 1999) or Repast Simphony (North et al. 2013), where, in addition to the developers themselves, the respective user communities perform verification of the provided simulation blocks and modules. Community members can not only detect bugs, but also correct them due to the open and collaborative nature of these projects. When such modelling toolkits are used, verification mainly consists of guaranteeing that the model has been correctly implemented using the available modules.

However, while the use of modelling toolkits reduces the programming and verification effort, it typically increases simulation times (Fachada et al. 2017a) and limits the modeller's flexibility in implementing non-standard behaviours (Sargent 2013). As such, it is often necessary to directly implement models using general-purpose programming languages. This is not a black or white choice, since several simulation toolkits offer the option of developing models using general-purpose programming languages (e.g. Repast Simphony), and/or provide high-performance and scalable workflows, with Repast HPC (Collier and North 2013) being a case in point.

When the direct use of general-purpose programming languages is involved, the adoption of good programming practices for designing and implementing the model is fundamental. Techniques such as object-oriented design, modularity and encapsulation not only simplify testing and debugging, but also promote incremental model development and the mapping of programming units (e.g. classes or functions) to model concepts, thus making computational models easier to understand, extend and modify. Additionally, defensive programming methodologies, such as assertions and unit tests, are well suited for the exploratory nature of simulation, making models easier to debug and verify.

Two important verification methods, *traces* and *structured walk-throughs*, complement the techniques discussed thus far. The former entails following a specific model variable (e.g. the position of an agent or the value of a simulation output) throughout the execution of the computational model, with the goal of assessing whether the implemented logic is correct and if the necessary precision is obtained. Modelling toolkits and programming language tools typically offer the relevant functionality, making the use of traces relatively simple (Sargent 2013). In turn, structured walk-throughs consist of having more than one person reading and debugging a program. All members of the development team are given a copy of a particular module to be debugged and the module developer goes through the code but does not proceed from one statement to the next until everyone is convinced that a statement is correct (Law 2015).

Nevertheless, and while the techniques described here are an important part of the verification process, a computational model should only be qualified as verified with reasonable confidence if it has been successfully replicated and/or aligned with a valid pre-existing model. We will return to this topic in greater detail in Sect. 9.4.

9.2.2 What Does It Mean to Validate a Model?

Model *validation* is defined as ensuring that both conceptual and computational models are adequate representations of the target. The term “adequate” in this sense may stand for a number of epistemological perspectives. From a practical point of view we could assess whether the outputs of the simulation are close enough to empirical data.

Alternatively, we could assess various aspects of the simulation, such as if the mechanisms specified in the simulation are well accepted by stakeholders involved in a participative-based approach. In Sect. 9.3 we will describe the general idea of validation as the process that assesses whether the pre-computational models—put forward as models of social complexity—can be demonstrated to represent theories or aspects of social behaviour able to give rise to post-computational models that are, at some given level, consistent with the onset theories or similar to real data.

Given the model development process described, is there any fundamental difference between verifying and validating simulations? Rather than being a sharp difference in kind it is a distinction that results from the computational method. Whereas verification is focused on the assessment of micro and macro concepts and inferences in the process of programming, observing and interpreting computational models, validation is focused on the evaluation of such inferences and concepts as representations of the target social phenomenon or theory.

In paraphrasing Axelrod (1997a), at first sight, we could say that the problem is whether an unexpected result is a reflection of the computational model, due to a mistake in the implementation of the pre-computational model, or is a surprising consequence of the pre-computational model itself. Unfortunately, the problem is more complicated than that. In many cases mistakes in the code may not be qualified simply as mistakes, but only as one interpretation among many others possible for implementing a conceptual model. Nevertheless, from a practical viewpoint there may be still good reasons to make the distinction between V&V. A number of established practices exist for the corresponding quadrants of Fig. 9.1. We will address some of these in the following sections.

9.3 Validation Approaches

We offered a conceptual definition of validation in Sect. 9.2.2. Had we given an operational definition, things would have become somewhat problematical. Models of social complexity are diverse and there is no definitive and guaranteed criterion of validity. As Amblard et al. (2007) remarked, "validation suggests a reflection on the intended use of the model in order to be valid, and the interpretation of the results should be done in relation to that specific context."

A specific use may be associated with different methodological perspectives for building the model, with different strategies, types of validity tests, and techniques (Fig. 9.2). Consider the kind of subjunctive, metaphorical models such as Schelling's (1971). In these models there is no salient validation step during the simulation development process. Design and validation walk together. The intended use is not to show that the simulation is plausible against a specific context of social reality but to propose abstract or schematic mechanisms as broad representations of classes of social phenomena. In other cases, the goal may be modelling a specific target domain, full of context, with use of empirical data and significant amounts of rich detail. Whereas in the former a good practice could be modelling with the greatest parsimony possible so as to have a computational model sanctionable by human beings and comparable to other models, parsimony can be in opposition to the goal of descriptive richness and thus inappropriate to the latter case.

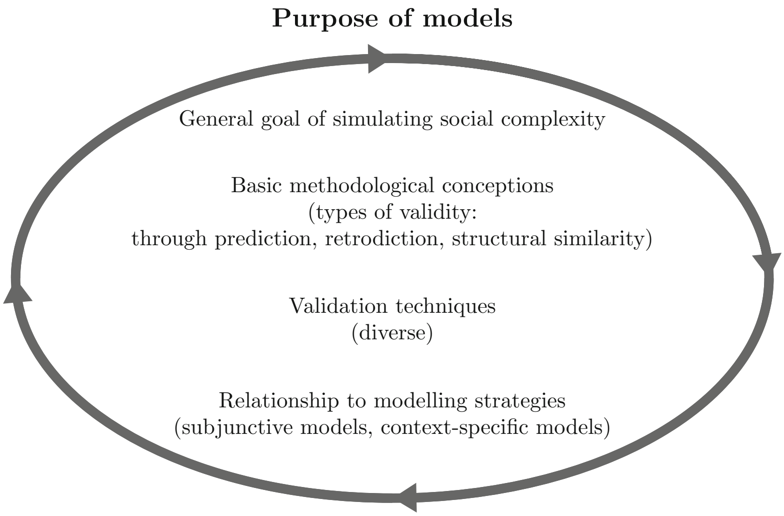

Fig. 9.2 Validation implies considering the purpose of the model

There are also different methodological motivations behind the use of a model, such as those conceived to predict or explain and those merely conceived to describe. Regardless of what method is used, the reproduction of characteristics of the object domain is important, but this can be assessed through rather different approaches during the model development process. If it is prediction you are seeking, validation consists of confronting simulated behaviour with the future behaviour of the target system (however, attempting to establish numerical prediction is not a normal goal in simulation). If it is explanation, validation consists of building plausible mechanisms that are able to reproduce simulated behaviour similar to real behaviour. If the goal is the more general aim of descriptiveness, explanation may probably be a goal as well, and a creative integration of ways for assessing the structure and results of the model, from quantitative to qualitative and participatory approaches, will be applied.

In conclusion, one should bear in mind that there is no one special method for validating a model. However, it is important to assess whether the simulation is subjected to good practices during its conception, whether it fits the intended use of the model builder and whether it is able to reproduce characteristics of the object domain. Assessing whether the goals of the modellers are well stated and the models themselves are well described in order to be understood and sanctioned by other model builders are *sine qua non* conditions for good simulation modelling.

In the remainder of this section, we revise the purpose of validating simulations along three dimensions: (1) the general goal of validation in social complexity; (2) basic methodological conceptions of validity types; and (3) typical validation techniques used in social simulation.

9.3.1 The Goal of Validation: Goodness of Description

If one is using a predictive model, then the purpose of the model is to predict either past or future states of the target system. On the other hand, one may strive for a model that is able to describe the target system with satisfactory accuracy in order to become more knowledgeable about the functioning of the system, to exercise future and past scenarios, and to explore alternative designs or inform policies.

The objective in this section is to define the purpose of validation in terms of the purpose of simulating social complexity, which we will define as being of good description. This position entails that there is no single method or technique for validating a simulation. A diversity of methods for validating models is generally applied.

In the rest of this chapter we adopt the agent-based paradigm for modelling. A conceptual understanding of validation, similar but more general than Moss and Edmonds (2005), will be used:

> The purpose of validation is to assess whether the design of micro-level mechanisms, put forward as theories of social complexity validated to arbitrary levels, can be demonstrated to represent aspects of social behaviour and interaction that are able to produce macro-level effects either (i) broadly consistent with the subjacent theories; and/or (ii) qualitatively or quantitatively similar to real data.

By *broad consistency* we mean the plausibility of both micro specification and macro effects accounted as general representations of the target social reality. In its most extreme expression, plausibility may be evaluated on a metaphorical basis. By *qualitative similarity* to real data we mean a comparison with the model in terms of categorical outcomes, accounted as qualitative features, such as the shape of the outcomes, general stylised facts, or dynamical regimes. As for *quantitative similarity* we mean the very unlikely case in which the identification of formal numerical relationships between aggregate variables in the model and in the target—such as enquiring as to whether both series may draw from the same statistical distribution—proves to be possible.

Notice that this definition is general enough to consider both the micro-level mechanisms and macro-level effects assessed on a participatory basis. It is also general enough to consider two methodological practices for building social simulation models, namely the extent to which models should be based on formal theories or on the intuition of the model builders and stakeholders—an issue that we will come back to later. These are omnipresent methodological questions in the social simulation literature and are by no means irrelevant to the purpose of simulation models.

Suppose that on the basis of a very abstract model, such as the Schelling model, you were to evaluate the similarity of its outputs with empirical data. Then you will probably not take issue with the fact that the goal of predicting future states of the target would be out of the scope of simulation research for that kind of modelling. However, despite the belief that other sorts of validation are needed, this does not imply excluding the role of prediction, but simply emphasises the importance of description as the goal of simulating social complexity. In truth, what could be more contentious in assessing the Schelling model is the extreme simplicity used to describe the domain of social segregation. The descriptive power of agent-based models (ABMs) makes them suited to model social complexity. Computational modelling corresponds to a process of abstraction, in that it selects some aspects of a subject being modelled, like entities, relations between entities and change of state, while ignoring those that may be considered less relevant to the questions that are of interest to the model builder. The expressiveness of ABMs allows the researcher to play with intuitive representations of distinct aspects of the target, such as defining societies with different kinds of agents, organisations, networks and environments, which interact with each other and represent social heterogeneity. By selecting certain aspects of social reality into a model, this process of demarcation makes agent-based modelling suited to represent sociality as perceived by researchers and often by the stakeholders themselves.

The descriptive power of simulation is on par with the diversity of ways used for informing the construction and validation of models, from theoretic approaches to the use of empirical data or stakeholder involvement. At any rate, measuring the goodness of fit between the model and real data expressed with data series is neither the unique nor a typical criterion for sanctioning a model. The very idea of using a diversity of formal and informal methods is to assess the credibility of the mechanisms of the model as good descriptions of social behaviour and interaction, which must be shown to be resilient in the face of multiple tests and methods, in order to provide robust knowledge claims and allow the model to be open to scrutiny.

9.3.2 Broad Types of Validity

When we speak about types of validity we mean three general methodological perspectives for assessing whether a model is able to reproduce expected characteristics of an object domain: validation through *prediction*, validation through *retrodiction* and validation through *structural similarity*. Prediction refers to validating a model by comparing the states of a model with future observations of the target system. Retrodiction compares the states of the model with past observations of the target system. Lastly, structural similarity refers to assessing the realism of the structure of the model in terms of empirical and/or theoretical knowledge of the target system (see also Gross and Strand 2000). In practice, all three approaches are interdependent and no single approach is used alone.

9.3.2.1 Validity Through Prediction

Validation through prediction requires matching the model with aspects of the target system before they were observed. The logic of predictive validity is the following: if one is using a *predictive model*—in which the purpose of the model is to predict future states of the target system—and the predictions prove satisfactory in repeated tested events, it may be reasonable to expect the model outcomes to stay reliable under similar conditions (Gross and Strand 2000). The purpose of prediction is somewhat problematic in social simulation:

– Models of social complexity usually show nonlinear effects in which the global behaviour of the model can become path-dependent and self-reinforcing, producing high sensitivity to initial conditions, which limits the use of predictive approaches.
– Many social systems show high volatility with unpredictable events, such as turning points of macroeconomic trade cycles or of financial markets that are in practice (and possibly in principle) impossible to predict; refer to Moss and Edmonds (2005) for a discussion on this.
– Many social systems are not amenable to direct observation, change too slowly, and/or do not provide enough data to be able to compare model outcomes. Most involve human beings and are too valuable to allow repeated intervention, which hinders the acquisition of knowledge about its future behaviour. Policies based on false predictions could have serious consequences, thus making the purpose of prediction unusable (Gross and Strand 2000).

While quantitative prediction of the target system behaviour is rare or simply unattainable, prediction in general is not able to validate per se the *mechanisms* of the model as good representations of the target system. In the words of Troitzsch (2004), "What simulations are useful to predict is only how a target system might behave in the future qualitatively." But a different model using different mechanisms that could lead to the same qualitative prediction may always exist, thus providing a different explanation for the same prediction. More often, the role of predicting future states of the target system becomes the exploration of new patterns of behaviour that were not identified before in the target system, whereby simulation acquires a speculative character useful as a heuristic and learning tool. What we are predicting is really new concepts that we had not realised as being relevant just by looking into the target.

9.3.2.2 Validity Through Retrodiction

The difference from retrodiction to prediction is that in the former the intention is to reproduce *already* observed aspects of the target system. Given the existence of a historical record of facts from the target system, the rationale of *retrodictive* validity for a *predictive* model is the following: If the model is able to reproduce a historical record consistently and correctly, then the model may also be trusted for the future (Gross and Strand 2000). However, as we have mentioned, predictive models of social complexity are uncommon in simulation. Explanation rather than prediction is the usual motive for retrodiction. The logic of retrodictive validity is the following: If a model is able to consistently reproduce a record of past behaviours of the target system, then the mechanisms that constitute the model are *eligible candidates* for explaining the functioning of the target system. Nevertheless, retrodiction alone is not sufficient to assess the validity of the candidate explanations:

- Underdetermination: Given a model able to explain a certain record of behaviours or historical data, there will always be a different model yielding a different explanation for the same record.
- Insufficient quality of data: In many cases it is impossible to obtain long historical series of social facts in the target system. In the social sciences the very notion of social facts or data is controversial, can be subjective, and is not dissociable from effects introduced by the measurement process. Moreover, even when data is available it may not be in a form suitable to be matched to the bulk of data generated by simulation models.

Underdetermination and insufficient data suggest the crucial importance of domain experts for validating the *mechanisms* specified in the model. A model is only valid provided that *both* the generated outcomes and the mechanisms that constitute the model are sanctioned by experts in the relevant domain. The importance of validating the mechanisms themselves leads us to the *structural* validity of the model, which neither predictive nor retrodictive validity is able to assess alone.

9.3.2.3 Validity Through Structural Similarity

In practice, the evaluation of a simulation includes some kind of prediction and retrodiction, based on expertise and experience. Given the implementation of micro-level mechanisms in the simulation, classes of behaviour at the macroscopic scale are identified in the model and compared to classes of behaviour identified in the target. Similarly, known classes of behaviour in the target system are checked for existence in the simulation. The former case is generally what we call the "surprising" character of simulations in which models show something beyond what we expect them to. However, only an assessment of the model from various points of view, including its structure and properties on different grains and levels, will truly determine whether it reflects the way in which the target system operates. For instance, do agents' behaviour, the constituent parts and the structural evolution of the model match the conception we have about the target system with satisfactory accuracy? These are examples of the elements of realism between the model and the system that the researcher strives to find, which requires expertise in the domain on the part of the person who builds and/or validates the model.

9.3.3 Validation Techniques

In this section we describe validation techniques used in social simulation. Some are used as common practices in the literature and most of the terminology has been inhered from simulation in engineering and computer science, particularly from the reviews of validation and verification in engineering by Sargent (2013). All techniques that we describe can be found in the literature, but it would be rare to find a model in which only one technique was used, consistent with the fact that the validation process should be diverse. Also, there are no standard names in the literature and some techniques overlap with others.

9.3.3.1 Face Validity

Face validity is a general kind of test used both before and after the model is put to use. During the model development process, the various intermediate models are presented to persons who are knowledgeable about the problem in order to assess whether they are compatible with the expert's understanding and reasonable for their purpose (Sargent 2013). Face validity may be used for evaluating the conceptual model, the components thereof, and the behaviour of the computational models in terms of categorical outcomes or direct input/output relationships. This can be accomplished via documentation, graphing visualisation models, and animation of the model as it moves through time. Visualisation of model outputs (including a brief look at model animation) is analysed in Chap. 10 of this volume (Evans et al. 2017). Insofar as this is a general kind of test, it is used in several iterations of the model.

9.3.3.2 Turing Tests

People who are knowledgeable about the behaviour of the target system are asked if they can discriminate between system and model outputs (Sargent 2013; Law 2015). The logic of Turing tests is the following: If the outputs of a computational model are qualitatively or quantitatively indistinguishable from the observation of the target system, a substantial level of validation has been achieved.

Note that the behaviour of the target system does not need to be observed directly in the cases where a computational direct representation is available. For example, suppose that videos of car traffic are transformed into three-dimensional scenes, whereby each object in the scene represents a car following the observed trajectory. If an independent investigator is not able to distinguish the computational reproduction from an agent-based simulation of car traffic, then a substantial level of validation has been obtained for the set of behaviours represented in the simulation model.

9.3.3.3 Historical Validity

Historical validity is a kind of retrodiction where the results of the model are compared with the results of previously collected data. If only a portion of the available historical data is used to design the model then a related concept is called *out-of-sample tests* in which the remaining data are used to test the predicative capacity of the model.

9.3.3.4 Event Validity

Event validity compares the occurrence of particular events in the model with the occurrence of events in the source data. This can be assessed at the level of individual trajectories of agents or at any aggregate level. Events are situations that should occur according to pre-specified conditions, although not necessarily predictable. Some events may occur at unpredictable points in time or circumstances. For instance, if the target system data shows arbitrary periods of stable behaviours interwoven with periods of volatility with unpredictable turning points, the simulation should produce similar kinds of unpredictable turning events.

9.3.3.5 Validity of Simulation Output

Since data is hard to collect in social systems, investigating the behaviour of simulation output becomes a crucial model validation technique (Sargent 2013). This can be performed by running the simulation with different parametrisations and checking if the output is reasonable (Law 2015), either based on subjective expert opinion when using "typical" simulations parameters, or by objectively evaluating output behaviour under trivial or extreme parametrisations. For instance, concerning the latter, if interaction among agents is nearly suppressed the modeller should be surprised if such activities as trade or culture dissemination continues in a population.

The concept of internal validity (Sargent 2013) or verification between the executable computational model and post-computational models (lower left quadrant of Fig. 9.1) can also be considered here, since it directly relates to simulation output behaviour. In order to assess the level of stochastic variability in a model, a number of simulation runs are performed using different random number streams. A sizeable level of variability between simulation runs can question the model at different levels. For example, the validity of simulation output for the executable computational model may be disputed, or the stability of a given policy (and the parametrisation that expresses it) in the overall model may be challenged.

For a more in-depth look at issues concerning simulation output behaviour, we refer the reader to the following references. Visualisation-oriented approaches for understanding simulation output are debated in Chap. 10 of this volume (Evans et al. 2017). Visualisation, and statistical and analytical analysis of model outputs are examined and reviewed by Lee et al. (2015). For a pure statistical outlook, Fachada et al. (2015) discuss a generic and systematic approach for evaluating time-series output of simulation models.

9.3.3.6 Solution Space Exploration

The techniques discussed in Sect. 9.3.3.5 are useful for basic output validation under specific parametrisations. However, they do not provide a general understanding of how input parameters influence model behaviour, nor they consider the broader picture of overall model assumptions, which encompass not only input parameters, but also internal model structure, employed submodels and model elements, as well as their inter-relations. In solution space exploration, model assumptions are varied in order to reach a better understanding of how the assumptions of interest affect the model.

The exploration of the solution space can be as simple as testing "what if" scenarios for observing model behaviour under different inputs—similar to what was discussed in the previous subsection—or follow a more systematic approach based on carefully designed experiments (Montgomery 2012). The latter approach aims to get the maximum amount of information from the model with the minimum number of simulation runs (Pereda et al. 2015), and is generally more efficient than hand-guided runs where alternative model configurations are experimented with (Law 2015). Nonetheless basic hand-guided experiments are also valuable for model validation, namely when trying different conceptual- or system-level assumptions. Conceptual-level assumptions include internal mechanisms or submodels that constitute the larger model (e.g. the decision processes of the agents, their learning mechanisms or their interaction topology), while system-level assumptions involve low-level elements of the model (e.g. agent activation regimes). If changing elements at the system-level determines different behaviours of the model that cannot be adequately interpreted, then the validity of the model can be compromised. The case of changing elements at conceptual levels is more subtle and the validity of the results must be assessed by the researcher with reference to the validity of the composing elements of the model. This is basically a kind of cross-model or cross-element validation, as described in Sect. 9.4.

The exploration of the solution space is often undertaken with one or more targeted objectives in mind, especially in the case of formally designed experiments. Typical objectives include optimisation, calibration, uncertainty analysis and sensitivity analysis (Lee et al. 2015). While these objectives may overlap, brief definitions and their potential roles in model validation can be given. In model *optimisation*, the researcher is interested in finding parameters or assumptions that minimise some cost or elicit specific model events or behaviour, which can be directly related with event validity, as discussed in Sect. 9.3.3.4. In turn, *calibration* is concerned with finding the assumptions that maximise the agreement of the model behaviour with the target system behaviour, thus making it a crucial aspect in model validation and in the model development process. *Uncertainty analysis* provides measures related to the reliability of results and how do input uncertainties propagate through to the collected outputs. These measures affect simulation output validity and directly influence the interpretation of data obtained through *sensitivity analysis*. The latter is arguably the most common objective when exploring the solution space of a model. In essence, small perturbations are applied to model assumptions in order to determine which ones have the greatest effect on output behaviour (Evans et al. 2017). This information can be used to improve model accuracy and reduce output variance—issues directly related with model validation—and also to promote model parsimony by fixing inconsequential parameters and simplifying assumptions, reducing dimensionality of the input parameter space and the model's computational cost (Law 2015; Lee et al. 2015). Conversely, sensitivity analysis may also point to underspecified assumptions, which may require additional detail in order to accurately represent some aspect of the target system (Law 2015). If the output remains unpredictable even with controlled changes, the modeller should be concerned about making claims about the model.

A number of techniques for sampling the solution space are described in the modelling and simulation literature. The *one-factor-at-a-time* (OFAT) approach is one of the simplest sampling techniques. The effects of individual assumptions (factors) on model behaviour are analysed in isolation by iterating each one over a set of discretised levels while keeping the other factors unchanged (Lee et al. 2015). Unfortunately, this technique ignores possible interactions between factors (Law 2015). This issue is handled by *factorial*-type designs, for which the different factor levels are combined in specific configurations (e.g. full factorial, fractional factorial or central composite designs) (Pereda et al. 2015). *Space-filling* designs are another type of sampling technique, and aim to cover the solution space more evenly (Pereda et al. 2015). Monte Carlo random sampling is probably the most common space-filling approach, consisting in sampling each parameter range randomly. However, care should be taken with this approach since clustered observations and empty spaces are bound appear by chance. Space-filling alternatives such as quasi-Monte Carlo or Latin Hypercube Sampling (McKay et al. 1979) cover the input space more evenly and are often preferred. In turn, sampling based on *meta-heuristics*, such as genetic algorithms, can search for pre-specified output behaviours. Thus, such techniques are commonly used when the researcher wishes to estimate parameters for calibration and/or optimisation purposes (Miller 1998; Calvez and Hutzler 2005; Stonedahl and Wilensky 2010).

Since the vast majority of the models of interest in social simulation are stochastic, one should also consider the issue of having to perform several runs with different seeds for each sampled assumption set in order to reduce the uncertainty about the expected output value. Consequently, there is a trade-off between assumption space coverage and output accuracy, which can severely limit the exploration of models with long execution times (Pereda et al. 2015). This issue can be minimised with the use of metamodels, which can act as computationally inexpensive proxies of more complex models (Lee et al. 2015). A metamodel, or a model of a model, can be used for predicting the original model's response for non-simulated assumption sets or finding combinations of assumptions that optimise (i.e. minimise or maximise) a response (Law 2015). A metamodel usually takes the form of a regression function relating inputs with an output response, typically a statistic representative of model behaviour. Statistical learning techniques such as regression analysis, Gaussian process modelling (Kriging), neural networks or random forests are commonly used for building the metamodel function (Law 2015; Pereda et al. 2015).

9.3.3.7 Participatory Approaches for Validation

Participatory approaches refer to the involvement of stakeholders both in the design and the validation of a model. Such an approach, also known as Companion Modelling (Barreteau et al. 2001), assumes that model development must be itself considered in the process of social intervention, where dialogue among stakeholders, including both informal and theoretical knowledge, is embedded in the model development process. Rather than just considering the final shape of the model, both the process and the model become instruments for negotiation and decision making. Documentation and visualisation techniques can play a crucial role in bridging the opinions and intentions of all interested parties. Such approaches are particularly suited for policy or strategy development. This topic is discussed in Chap. 12 “Participatory Approaches” (Barreteau et al. 2017).

9.4 Replicating and Comparing Models

Computational models in social science can be very sensitive to implementation details, and the influence that seemingly negligible aspects such as data structures or sequences of events can have on simulation results is striking (Merlone et al. 2008). Furthermore, model implementations can be considerably elaborate, making them prone to programming errors (Will and Hegselmann 2008). This can seriously affect V&V when data from the system being modelled cannot be obtained easily, cheaply or at all—often the case in social simulation. Moreover, even if data were available, the goodness of fit between real and simulated data, albeit reflecting evidence about the validity of the model as a data-generating process, does not provide evidence on how it operates. Model replication—the reimplementation of an existing model and the replication of its results—is a potential but frequently neglected solution to this problem (Will and Hegselmann 2008; Thiele and Grimm 2015). Replicating a model in a different context will sidestep the biases associated with the language or toolkit used to develop the original model, bringing to light inconsistencies between conceptual and computational models (Edmonds and Hales 2003; Wilensky and Rand 2007).

Replication strongly contributes to the V&V of simulation models (Wilensky and Rand 2007; Thiele and Grimm 2015). Verification is improved because if two or more distinct implementations of a conceptual model yield equivalent results, it is more likely that the implemented models correctly describe the conceptual model (Wilensky and Rand 2007). In turn, validation is stimulated since its very idea is comparing models with other descriptions of the problem modelled, and this may include *cross-model validation*, i.e. the comparison with other simulation models that have been validated to some level. Thus, it is reasonable to assume that a computational model cannot be considered fully verified and validated until it has been successfully replicated (Edmonds and Hales 2003). Nonetheless, the most important reason for replicating and comparing models is simply one of good scientific practice, since replication is the gold standard against which scientific claims are evaluated (Peng 2011).

In the remainder of this section we discuss replication and comparison of simulation models under three different perspectives. First, in Sect. 9.4.1, we distinguish the terminology and origins of the different goals related to model replication and comparison. Next, in Sect. 9.4.2, we go over the best practices in developing models so that they may be replicated by other researchers in the future. Finally, in Sect. 9.4.3 we discuss a number of model comparison techniques.

9.4.1 Model Replication, Model Alignment or Submodel Comparison?

A *model replication* study commonly assesses the extent to which building computational models that draw on the same conceptual, usually published, model give results compatible with the ones reported for the latter. If the new results are similar to the published results, then the confidence in the correspondence between the computational and the conceptual models is increased. Replication is represented in Fig. 9.3.

The work of Edmonds and Hales (2003) is particularly informative and worthy of reference. Edmonds and Hales performed two independent replications of a previously published model involving co-operation between self-interested agents. Several shortcomings were found in the original model, leading the authors to conclude that unreplicated simulation models and their results cannot be trusted. The issue was found to be a subtle difference in one of the submodels, which lead to different conclusions about the functioning of the overall model.

The term *model alignment* is frequently used as a synonym for model replication. However, its meaning is somewhat more subtle, as it is more related with the extent to which models can be coupled or docked so that their consequences and results are consistent with each other. In its most general form, this concerns both to V&V. After Axtell et al. (1996), the term became associated with the process of determining whether different published models describing the same class of social phenomena produce the same results. Usually the alignment or docking of

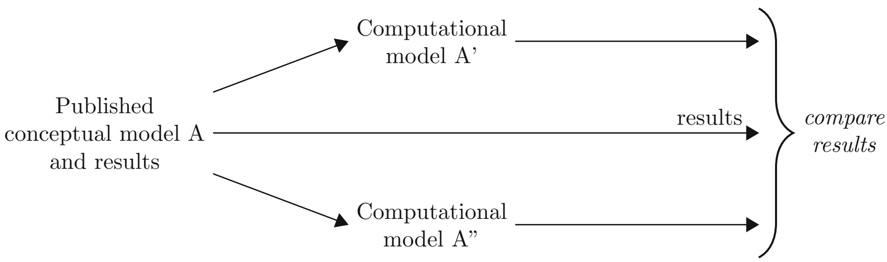

Fig. 9.3 Model replication

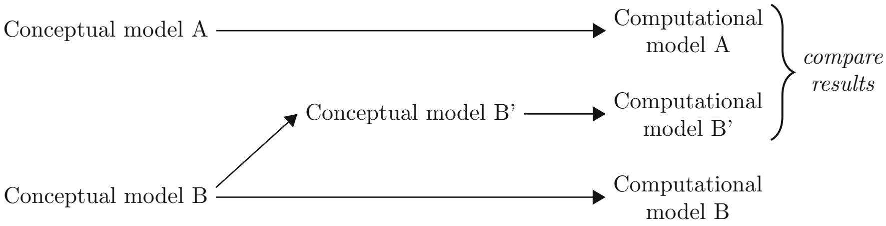

Fig. 9.4 Model alignment, also referred to as docking

two models A and B requires modifying certain aspects of model B—for instance turning off a specific feature—in order to become equivalent to model A. This is represented in Fig. 9.4.

The work of Axtell et al. (1996) is arguably the most-cited attempt to align two distinct but similar models. Rather than re-implementing Axelrod's culture dissemination model, Axtell and colleagues focused on the general case of aligning two models that reflected slightly distinctive mechanisms. For this purpose, Epstein and Axtell's Sugarscape model (1996) was progressively simplified in order to align with the results obtained by Axelrod's culture dissemination model (1997b). They concluded that comparing models developed by different researchers and with different tools (i.e. programming languages and/or modelling environments), can lead to exposing bugs, misinterpretations in model specification, and implicit assumptions in toolkit implementations.

Model alignment has been further investigated in a series of meetings called model-to-model (M2M) workshops (Rouchier et al. 2008). The M2M workshops attract researchers interested in understanding and promoting the transferability of knowledge between model users.

Submodel comparison, often referred to as *cross-element validation*, rather than comparing whole models, compares the results of a model whose architecture of the agents differs only in a few elements. The goal is to assess the extent to which changing elements of the model architecture produces results compatible with the expected results of the (larger) model. It is essentially an exercise in composing different submodels within a larger model, and is related to solution space exploration since submodels are varied, and the consequences of that variation are analysed and compared. In this process, the overall validity of the larger model with reference to the validity of each one of the submodels can also be assessed. For instance, one may study the effects of using a model with agents in a bargaining game employing either evolutionary learning or reinforcement learning strategies, and assess which one of the strategies produces results compatible with theoretical analysis in game theory (Takadama et al. 2003). Submodel comparison is represented in Fig. 9.5.

Submodel comparison can also be used as a model replication or alignment aid. For example, Radax and Rengs (2009) proposed a method for replicating insufficiently described ABMs, consisting in systematically varying ambiguous

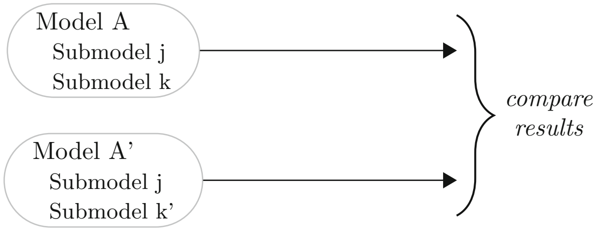

Fig. 9.5 Submodel comparison, also referred to as cross-element validation

model elements in order to align the replicated model with the original one. More generally, if two simulations do not align, trying out different assumptions or submodels is a practical way of finding the source of errors or mismatches. This type of study is greatly facilitated when computational models are implemented in a modular fashion, as discussed in Sect. 9.2.1. If submodels or model elements are implemented as separate modules in the computational model, it becomes much simpler to change or swap them in order to perform submodel comparisons.

9.4.2 Developing Replicable Models

An important aspect when developing a simulation model is to guarantee that it may be replicated by other researchers. Designing and programming for replicability involves a number of aspects that should be considered. Simulations are often a mix of conceptual descriptions and hard technical choices about implementation. The author who reports a model should assume that a replication or alignment may later be tried and thus should be careful about providing detailed information for future use. Some of the best practices include, but are not limited to:

– Effective documentation about the conceptual model should be provided, preferably in the form of a structured natural language description (Müller et al. 2014), such as the ODD protocol, discussed in Chap. 15 (Grimm et al. 2017) of this volume.The ODD protocol (Overview, Design concepts, Details) is one of the most widely used templates for making model descriptions more understandable and complete, providing a comprehensive checklist that covers many of the key features that can define a model. The ODD + D protocol (Müller et al. 2013) extends the ODD protocol for models in which human-decision making is simulated, often the case in social simulation.
– The model's source code should be made available, given that it is the model's definitive implementation, not subject to the vagueness and uncertainty possibly associated with verbal descriptions (Wilensky and Rand 2007; Müller et al. 2014). If possible, an open source simulation platform should be utilised to implement the model, thus fostering software reuse in order to make simulations

reliable and more comparable to each other. Maximum model exposure is achieved if the simulation is runnable on the browser. This is much simpler nowadays, with technologies such as HTML5 and JavaScript dispensing the need for browser plug-ins. ABM toolkits such as AgentScript (Densmore 2016) and AgentBase (Wiersma 2015) use this approach. In any case, making the computational model widely available and easily runnable is crucial for others to be able to experiment with it.
– Besides source code availability, documentation about the computational model should also be provided in the form of (1) detailed source code comments, and (2) a user guide and/or technical report. The former should clearly explain what each code unit (e.g. function or class) does, while the latter should describe the program's architecture, preferably with the aid of visual description standards such as UML diagrams. In either case, the computational model documentation should contain information about technical options where the translation from the conceptual model was neither straightforward nor consensual.
– Detailed information about the results should be made publicly available. This includes statistical methods and/or scripts implementing or using them, raw simulation outputs, distributional information, sensitivity analyses performed or qualitative measures. A number of specialised scientific data repositories exist for this purpose (Assante et al. 2016; Amorim et al. 2015). Furthermore, there is an increasing awareness of how important it is to have published, citable and documented data available in the scholarly record due to its crucial role in reproducible science (Altman et al. 2015; Kratz and Strasser 2014).

The CoMSES Net Computational Model Library (Rollins et al. 2014), an open digital repository for disseminating computational models associated with publications in the social and life sciences, should be highlighted in this regard since it enforces some of the best practices discussed above. Models are organised as searchable entries, by title, author or other relevant metadata. A formatted citation is shown for each entry so that researchers who use the model can easily credit its creators. Model entries have separate sections for code, documentation, generated outputs, solution exploration analyses and other relevant information. The library accepts not only original models, but also explicitly welcomes replications of previous studies. It also offers a certification service that verifies (1) if the model code successfully compiles and runs, and (2) if the model adheres to documentation best practices, with the ODD protocol being the recommended documentation template.

9.4.3 Model Comparison Techniques

Replication is evaluated by comparing the outputs of the original computational model against the output of the replicated implementation (Thiele and Grimm 2015). However, how do we determine whether or not two models produce equivalent output behaviour? Axtell et al. (1996) defined three kinds of equivalence or levels of similarity between model outputs: *numerical identity*, *relational alignment* and *distributional equivalence*. The first, *numerical identity*, implies exact numerical output and is difficult to demonstrate for stochastic models in general and social complexity models in particular. *Relational alignment* between outputs exists if they show qualitatively similar dependencies with input data, which is frequently the only way to compare a model with another which is inaccessible (e.g. implementation has not been made available by the original author), or with a non-controllable "real" social system. Lastly, *distributional equivalence* between implementations is achieved when the distributions of results cannot be statistically distinguished. What this shows is that at conventional confidence probabilities the statistics from different implementations may come from the same distribution, but it does not prove that this is actually the case. In other words, it does not prove that two implementations are algorithmically equivalent. Nonetheless, demonstrating equivalence for a larger number of parametrisations increases the confidence that the implementations are in fact globally equivalent (Edmonds and Hales 2003).

Since numerical identity is difficult to attain, and is not critical for showing that two such models have the same dynamic behaviour, distributional equivalence is more often than not the appropriate standard when comparing two implementations of a stochastic social complexity model. When aiming for distributional equivalence, a set of statistical summaries representative of each output are selected. It is these summaries, and not the complete outputs, that will be compared in order to assess the similarity between the original computational model and the replicated one. As models may produce large amounts of data, the summary measures should be chosen as to be relevant to the actual modelling goal. The summaries of all model outputs constitute the set of focal measures (FMs) of a model (Wilensky and Rand 2007), or more specifically, of a model parametrisation (since different FMs may be selected for distinct parametrisations). However, this process is empirically driven and model-dependent, or even parameter-dependent. Furthermore, it is sometimes unclear as to what output features best describe model behaviour. A possible solution, presented by Arai and Watanabe (2008) in the context of comparing models with different elements, is the automatic extraction of FMs from time-series simulation output using the discrete Fourier transform. Fachada et al. (2017b) proposed a similarly automated method, using principal component analysis to convert simulation output into a set of linearly uncorrelated statistical measures, analysable in a consistent, model-independent fashion. The proposed method was broader in scope—with support for multiple outputs and different types of data—and is available in the form of a software package for the R platform (Fachada et al. 2016; R Core Team 2017).

Once the FMs are extracted from simulation output, there are three major statistical approaches used to compare them: (1) statistical hypothesis tests; (2) confidence intervals; and (3) graphical methods (Balci and Sargent 1984). Statistical hypothesis tests are often used for comparing two or more computational models (Axtell et al. 1996; Wilensky and Rand 2007; Edmonds and Hales 2003; Miodownik et al. 2010; Radax and Rengs 2009; Fachada et al. 2017a,b). More specifically, hypothesis tests check if the statistical summaries obtained from the outputs of two (or more) model implementations are drawn from the same distribution. Confidence intervals are usually preferred for comparing the output of a model with the output of the system being modelled, as they provide an indication of the magnitude by which the statistic of interest differs between the two. Nonetheless, confidence intervals can also be used for model comparison, but in contexts different from replication, such as the evaluation of different models that might represent competing system designs or alternative operating policies (Balci and Sargent 1984; Law 2015). Graphical methods, such as Q–Q plots (e.g. Alberts et al. 2012) or scatter plots (e.g. Arai and Watanabe 2008; Fachada et al. 2017b), can also be employed for comparing output data, though their interpretation is more subjective than the previous methods.

9.5 Modelling Strategies and Its Relationship to Validation

In this section we review the purpose of validation and its relationship to different modelling strategies with respect to the level of descriptive detail embedded in a simulation.

Several taxonomies of modelling strategies have been described in the literature (David et al. 2004; Boero and Squazzoni 2005; Gilbert 2008, pp. 42–44). Normally, the adoption of these strategies does not depend on the class of the target being modelled, but on different ways to address it as the problem domain. For example, if a simulation is intended to model a system for the purpose of designing policies, this implies representing more information and detail than a simulation intended for modelling social mechanisms of the system in a metaphorical way. However, varying levels of model detail imply a trade-off between the effort required for verifying the simulation and the effort required for validating it. As more context and richness are embedded in a model, the more difficult it will be to verify it. Conversely, as one increases the descriptive richness of simulations, more ways will be available to assess its validity. A tension that contrasts the tendency for constraining simulations by formal-theoretical constructs—normally easier to verify—and constraining simulations by theoretical-empirical descriptions—more amenable to validation by empirical and participative-based methods. In the next sections, two contrasting modelling strategies are discussed and the typical cycle of formal and informal approaches for modelling and validation is described.

9.5.1 Subjunctive Agent-Based Models

A popular strategy in social simulation consists of using models as a means for expressing subjunctive moods to talk about possible worlds using what-if scenarios, like "what would happen if something were the case." The goal is building artificial societies for modelling possible worlds that represent classes of social mechanisms, while striving for maximal simplicity and strong generalisation power of the representations used. Reasons for striving for simplicity include the computational tractability of the model and to keep the data analysis as simple as possible.

Simplicity and generalisation power are often seen as elements of elegance in a model. However, making the model simpler in the social sciences does not necessarily make the model more general. More often than not this kind of modelling only makes it metaphorically general, or simply counterfactual (with false assumptions). For example, "What would happen if world geography is regarded as a two-dimensional space arranged on a 10×10 grid, where agents are thought of as independent political units, such as nations, which have specific behaviours of interaction according to simple rules?" To assume that world geography is one-dimensional, as Axelrod (1993) does in his Tribute Model, is clearly a false assumption. Often these models are associated with a design slogan coined by Axelrod (1997a), called the KISS approach—"Keep it Simple Stupid." Despite their simplicity, these kinds of models prove useful for concept formation and theoretical abstraction. The emergence of macro regularities from micro-levels of interaction becomes the fundamental source of concept formation and hypothesis illustration, with the power of suggesting novel theoretical debates.

Given the tendency for simplification and abstraction, mechanisms used in these models are normally described in a formalised or mathematical way. Axelrod's models, such as the culture dissemination model, or Schelling's residential segregation model, are canonical examples. Their simplicity and elegance have been factors for popularity and dissemination that span numerous disciplines and ease replication and verification.

However, whereas simplicity eases verification, the use of metaphorical models also brings disadvantages. Consider a word composed of several attributes representing an agent's culture, such as in Axelrod's culture dissemination model. The attributes do not have any specific meaning and are only distinguishable by their relative position in the word. Thus, they can be interpreted according to a relatively arbitrary number of situations or social contexts. However, such a representation may also be considered too simplified to mean anything relevant for such a complex concept as a cultural attribute. As a consequence, verification is hardly distinguishable from validation, insofar as the model does not represent a specific context of social reality. In such a sense, the researcher is essentially verifying experimentally whether his conceptions are met by an operationalisation that is intentionally and computationally expressed (David et al. 2005). Nevertheless, given their simplicity, subjunctive models can be easily linked and compared to other models, extended with additional mechanisms, as well as modified for model alignment, docking, or replication. Cross-element validation is a widely used technique.

At any rate, the fact that these models are simpler to replicate and compare—but hardly falsifiable by empirically acquired characteristics of social reality—stresses their strong characteristic: when models based on strategies of maximal simplicity become accepted by a scientific community, their influence seems to reach several other disciplines and contexts. Perhaps for this reason, these kinds of models are the most popular in social simulation, and some models are able to reach a considerable impact in many strains of social science.

9.5.2 Context-Specific Agent-Based Models

It would be simplistic to say that models in social simulation can be characterised according to well-defined categories of validation strategies. Even so, the capacity to describe social complexity, whether through simplicity or through rich detail and context, is a determining factor for a catalogue of modelling strategies.

We cannot hope to model general social mechanisms that are valid in all contexts. There are many models that are not designed to be markedly general or metaphorically general, but to stress accurateness, diversity, and richness of description. Instead of using possible worlds representing very arbitrary contexts, models are explicitly bounded to specific contexts. Constraints imposed on these models can vary from models investigating properties of social mechanisms in a large band of situations which share common characteristics, to models with the only ambition of representing a single history, like Dean's retrodiction of the patterns of settlement of the Anasazi in the southwestern United States, household by household (Dean et al. 2000).

Constructing and validating a model of this kind requires the use of empirical knowledge. They are, for this reason, often associated with the idea of "Empirical Validation of Agent-Based Models."

What is the meaning of empirical in this sense? If the goal is to discuss empirical claims, then models should attempt to capture empirically enquired characteristics of the target domain. Specifying the context of descriptions will typically provide more ways for enquiring quantitative and qualitative data in the target, as well as using experimental and participative methods with stakeholders. In this sense, empirical may be understood as a stronger link between the model and a context-specific, well-circumscribed problem domain.

The Anasazi model by Dean et al. (2000) is a well-known and oft-cited example of a highly contextualised model built on the basis of numerous sources, from archaeological data to anthropological, agricultural and ethnographic analyses, in a multidisciplinary context.

Given the higher specificity of the target domain, the higher diversity of ways for enriching the model as well as the increased semantic specificity of the outputs produced by the model, context-specific models may be more susceptible to be compared with empirical results of other methods of social research. On the other hand, comparison with other simulation models is complex and these models are more difficult to replicate and compare.

9.5.3 *Modus Operandi: Formal and Informal Approaches*

The tension between simplicity and descriptive richness expresses two different ways for approaching the construction and validation of a model. One can start with a rich, complex, realistic description and only simplify it where this turns out to be possible and irrelevant to the target system—known as the KIDS approach (Edmonds and Moss 2005). Or one starts from the outset with the simplest possible description and complexifies it only when it turns out to be necessary to make the model more realistic (Law 2015), nevertheless keeping the model as simple as possible—known as the KISS approach (Axelrod 1997a).

In practice, both trends are used for balancing trades-offs between the model's descriptive accuracy and the practicality of modelling, according to the purpose and the context of the model (Sun et al. 2016). This raises yet another methodological question: the extent to which models ought to be designed on the basis of formal theories, or ought to be constrained by techniques and approaches just on the basis of the intuition of the model builders and stakeholders. As we have seen, strong, subjunctive, ABMs with metaphorical purposes tend to adopt the simplicity motto with extensive use of formal constructs, making the models more elegant from a mathematical point of view, easier to verify, but less liable to validation methods. Game theoretical models, with all their formal and theoretical apparatus, are a canonical example. Results from these models are strongly constrained by the formal theoretical framework used.

A similar problem is found when ABMs make use of cognitive architectures strongly constrained by logic-based formalisms, such as the kind of formalisms used to specify BDI-type architectures. If the cognitive machinery of the agents relies on heuristic approaches that have been claimed valid, many researchers in the literature claim that cognitive ABMs can be validated in the empirical sense of context-specific models. Cited examples of this kind usually point to ABMs based on the Soar cognitive architecture (Laird 2012).

At any rate, context-specific models are normally more eclectic and make use of both formal and informal knowledge, often including informal and stakeholder evidence in order to build and validate the models. Model design tends to be less constrained a priori by formal constructs. In principle, one starts with all aspects of the target domain that are assumed to be relevant and then explores the behaviour of the model in order to find out if there are aspects that do not prove relevant for a particular interval of outcomes. The typical approach the majority of all modelling and validation can be summarised in a cycle with the following iterative and overlapping steps:

(a) *Building and validating pre-computational and computational models*: Several descriptions and specifications are used to build a model, eventually in the form of a computer program, which are micro-validated against a theoretical framework and/or empirical knowledge, usually qualitatively. This may include the individual agents' interaction mechanisms (rules of behaviour for agents or organisations of agents), their internal mechanisms (e.g. their cognitive machinery), the kind of interaction topology or environment, and the passive entities with which the agents interact. The model used should be as general as possible for the context in consideration as well as flexible for testing how parameters vary in particular circumstances. Empirical data—if available—should be used to help configure the parameters. Both the descriptions of the model and the parameters used should be validated for the specific context of the model. For example, suppose empirical data are available for specifying the consumer demand of products. If the demand varies from sector to sector, one may use data to inform the distribution upon which the parameter could be based for each specific sector.

(b) *Specifying expected behaviours of the computational model*: Micro and macro characteristics that the model is designed to reproduce are established from the outset based on theoretical and/or empirical knowledge. Any property, from quantitative to qualitative measures, such as emergent key facts the model should reproduce (stylised facts), the statistical characteristic or shape of time-data series (statistical signatures) and individual agents' behaviour along the simulation (individual trajectories), can be assessed. This may be carried out in innumerable ways, according to different levels of description or grain, and be more or less general depending on the context of the model and the kind of empirical knowledge available. For instance, in some systems it may be enough to predict just a "weak" or "positive" measure on some particular output, such as a positive and weak autocorrelation. Or we might look for the emergence of unpredictable events, such as stable regimes interleaved with periods of strong volatility, and check their statistical properties for various levels of granularity. Or the emergence of different structures or patterns associated with particular kinds of agents, such as groups of political agents with "extremist" or "moderate" individuals.

(c) *Testing the computational model and building and validating post-computational models*: The computational model is executed. Both individual and aggregate characteristics are computed and tested for sensitivity analysis. These are micro-validated and macro-validated against the expected characteristics of the model established in step B according to a variety of validation techniques, as described in the previous sections. A whole process of building post-computational models takes place, possibly leading to the discovery of unexpected characteristics in the behaviour of the computational model which should be assessed with further theoretical or empirical knowledge about the problem domain.

Further Reading

Good introductions to validation and verification of simulation models in general are Sargent (2013) and Troitzsch (2004), the latter with a focus on social simulation. Validation of ABMs in particular is addressed by Amblard et al. (2007).

For readers more interested in single aspects of V&V, with regard to ABMs with applicability in social simulation, the following papers provide highly accessible starting points:

– Edmonds and Hales (2003) demonstrate the importance of model replication (or model alignment) by means of a clear example.
– Boero and Squazzoni (2005) examine the use of empirical data for model calibration and validation and argue that "the characteristics of the empirical target" influence the choice of validation strategies.
– Moss and Edmonds (2005) discuss an approach for cross-validation that combines the involvement of stakeholders to validate the model qualitatively on the micro level with the application of statistical measures to numerical outputs to validate the model quantitatively on the macro level.
– Müller et al. (2014) address the question of whether an ideal standard for describing and documenting models exists, defining different types of model reporting and proposing a minimum description standard for good modelling practice.
– Lee et al. (2015) provide an overview of the state-of-the-art approaches in analysing and reporting ABM outputs, highlighting challenges and issues related to variance stability, sensitivity analysis, spatio-temporal analysis, visualisation, and effective communication of these to non-technical audiences, such as various stakeholders.
– Fachada et al. (2017b) Present a structured approach to designing and performing complete model comparison experiments, using statistical tests to determine if two or more computational models generate distributionally equivalent behaviour.
– Finally, more comprehensive epistemological perspectives on verification and validation are provided in a number of papers published or derived from the Epistemological Perspectives on Simulation (EPOS) workshops, namely Frank and Troitzsch (2005), David (2009), Squazzoni (2009) and David et al. (2010).

Acknowledgements This work was partially funded by the Fundação para a Ciência e a Tecnologia project UID/EEA/50009/2013.

References

Alberts, S., Keenan, M. K., D'Souza, R. M., & An, G. (2012). Data-parallel techniques for simulating a mega-scale agent-based model of systemic inflammatory response syndrome on graphics processing units. *Simulation, 88*(8), 895–907. doi:10.1177/0037549711425180, http://journals.sagepub.com/doi/abs/10.1177/0037549711425180

Altman, M., Borgman, C., Crosas, M., & Matone, M. (2015). An introduction to the joint principles for data citation. *Bulletin of the Association for Information Science and Technology, 41*(3), 43–45. doi:10.1002/bult.2015.1720410313, http://onlinelibrary.wiley.com/doi/10.1002/bult.2015.1720410313/abstract

Amblard, F., Bommel, P., & Rouchier, J. (2007). Assessment and validation of multi-agent models. In *Agent-based modelling and simulation in the social and human sciences* (pp. 93–116). Oxford: Bardwell Press. http://agritrop.cirad.fr/541339/

Amorim, R. C., Castro, J. A., Silva, J. Rd., & Ribeiro, C. (2015). A comparative study of platforms for research data management: Interoperability, metadata capabilities and integration potential. In *New contributions in information systems and technologies* (pp. 101–111). Cham: Springer. doi:10.1007/978-3-319-16486-1_10, https://link.springer.com/chapter/10.1007/978-3-319-16486-1_10

Arai, R., & Watanabe, S. (2008). A quantitative method for comparing multi-agent-based simulations in feature space. In *Multi-agent-based simulation IX* (pp.154–166). Berlin/Heidelberg: Springer. doi:10.1007/978-3-642-01991-3_12, https://link.springer.com/chapter/10.1007/978-3-642-01991-3_12

Assante, M., Candela, L., Castelli, D., & Tani, A. (2016). Are scientific data repositories coping with research data publishing? *Data Science Journal, 15*, 6. doi:10.5334/dsj-2016-006, http://datascience.codata.org/articles/10.5334/dsj-2016-006/

Axelrod, R. (1993). *A Model of the Emergence of New Political Actors*. Working paper 93-11-068, Santa Fe Institute. https://www.santafe.edu/research/results/working-papers/a-model-of-the-emergence-of-new-political-actors

Axelrod, R. (1997a). Advancing the art of simulation in the social sciences. In D. R. Conte, P. D. R. Hegselmann, & P. D. P. Terna (Eds.), *Simulating Social Phenomena*. Lecture notes in economics and mathematical systems (Vol. 456, pp. 21–40). Berlin/Heidelberg: Springer. doi:10.1007/978-3-662-03366-1_2, http://link.springer.com/chapter/10.1007/978-3-662-03366-1_2

Axelrod, R. (1997b). The dissemination of culture: A model with local convergence and global polarization. *Journal of Conflict Resolution, 41*(2), 203–226. doi:10.1177/0022002797041002001, http://dx.doi.org/10.1177/0022002797041002001

Axtell, R., Axelrod, R., Epstein, J. M., & Cohen, M. D. (1996). Aligning simulation models: A case study and results. *Computational & Mathematical Organization Theory, 1*(2), 123–141. doi:10.1007/BF01299065, https://link.springer.com/article/10.1007/BF01299065

Balci, O., & Sargent, R. G. (1984). Validation of simulation models via simultaneous confidence intervals. *American Journal of Mathematical and Management Sciences, 4*(3–4), 375–406. doi:10.1080/01966324.1984.10737151, http://dx.doi.org/10.1080/01966324.1984.10737151

Barreteau, O., Bots, P., Daniell, K., Etienne, M., Perez, P., Barnaud, C., et al. (2017). Participatory approaches. doi:https://doi.org/10.1007/978-3-319-66948-9_12.

Barreteau, O., Bots, P., Daniell, K., Etienne, M., Perez, P., Barnaud, C., et al. (2017). Participatory approaches. In B. Edmonds & R. Meyer (Eds.), *Simulating social complexity. Understanding complex systems* (2nd ed.). Berlin/Heidelberg: Springer. doi:10.1007/978-3-319-66948-9_12

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. http://jasss.soc.surrey.ac.uk/8/4/6.html

Calvez, B., & Hutzler, G. (2005). Automatic tuning of agent-based models using genetic algorithms. In *Multi-agent-based simulation VI* (pp. 41–57). Berlin/Heidelberg: Springer. doi:10.1007/11734680_4, https://link.springer.com/chapter/10.1007/11734680_4

Collier, N., & North, M. (2013). Parallel agent-based simulation with repast for high performance computing. *Simulation, 89*(10), 1215–1235. doi:10.1177/0037549712462620, http://journals.sagepub.com/doi/abs/10.1177/0037549712462620

David, N. (2009). Validation and verification in social simulation: patterns and clarification of terminology. In *Epistemological aspects of computer simulation in the social sciences* (pp. 117–129). Berlin/Heidelberg: Springer. doi:10.1007/978-3-642-01109-2_9, https://link.springer.com/chapter/10.1007/978-3-642-01109-2_9

David, N., Marietto, M. B., Sichman, J. S., & Coelho, H. (2004). The structure and logic of interdisciplinary research in agent-based social simulation. *Journal of Artificial Societies and Social Simulation, 7*(3), 4. http://jasss.soc.surrey.ac.uk/7/3/4.html

David, N., Sichman, J. S., & Coelho, H. (2005). The logic of the method of agent-based simulation in the social sciences: Empirical and intentional adequacy of computer programs. *Journal of Artificial Societies and Social Simulation, 8*(4), 2. http://jasss.soc.surrey.ac.uk/8/4/2.html

David, N., Caldas, J. C., & Coelho, H. (2010). Epistemological perspectives on simulation III. *Journal of Artificial Societies and Social Simulation, 13*(1). doi:10.18564/jasss.1591, http://jasss.soc.surrey.ac.uk/13/1/14.html

Dean, J. S., Gumerman, G. J., Epstein, J. M., Axtell, R. L., Swedlund, A. C., Parker, M. T., et al. (2000). Understanding Anasazi culture change through agent-based modeling. In T. A. Kohler & G. J. Gumerman (Eds.), *Dynamics in human and primate societies: Agent-based modeling of social and spatial processes. Santa fe institute studies on the sciences of complexity* (pp. 179–205). New York/Oxford: Oxford University Press.

Densmore, O. (2016). AgentScript. http://agentscript.org/

Edmonds, B., & Hales, D. (2003). Replication, replication and replication: Some hard lessons from model alignment. *Journal of Artificial Societies and Social Simulation, 6*(4), 11. http://jasss.soc.surrey.ac.uk/6/4/11.html

Edmonds, B. & Moss, S. (2005). From KISS to KIDS—an 'anti-simplistic' modelling approach. In: P. Davidsson, B. Logan, & K. Takadama (Eds.), *Multi-agent and multi-agent-based simulation* (Vol. 3415, pp. 130–144). Berlin/Heidelberg: Springer. doi:10.1007/978-3-540-32243-6_11. http://link.springer.com/10.1007/978-3-540-32243-6_11

Epstein, J., & Axtell, R. (1996). *Growing artificial societies: Social science from the bottom up*. Washington, DC: Brookings Institution Press; Cambridge, MA: MIT Press.

Evans, A., Heppenstall, A., & Birkin, M. (2017). Understanding simulation results. doi: https://doi.org/10.1007/978-3-319-66948-9_10.

Fachada, N., Lopes, V. V., Martins, R. C., & Rosa, A. C. (2015). Towards a standard model for research in agent-based modeling and simulation. *PeerJ Computer Science, 1*, e36. doi:10.7717/peerj-cs.36, https://peerj.com/articles/cs-36

Fachada, N., Rodrigues, J., Lopes, V. V., Martins, R. C., & Rosa, A. C. (2016). micompr: An R package for multivariate independent comparison of observations. *The R Journal, 8*(2), 405–420. http://journal.r-project.org/archive/2016-2/fachada-rodrigues-lopes-etal.pdf

Fachada, N., Lopes, V. V., Martins, R. C., & Rosa, A. C. (2017a). Parallelization strategies for spatial agent-based models. *International Journal of Parallel Programming, 45*(3), 449–481.

Fachada, N., Lopes, V. V., Martins, R. C., & Rosa, A. C. (2017b). Model-independent comparison of simulation output. *Simulation Modelling Practice and Theory, 72*, 131–149. doi:10.1016/j.simpat.2016.12.013, http://www.sciencedirect.com/science/article/pii/S1569190X16302854

Frank, U., & Troitzsch, K. G. (2005). Epistemological perspectives on simulation. *Journal of Artificial Societies and Social Simulation, 8*(4), 7. http://jasss.soc.surrey.ac.uk/8/4/7.html

Galán, J. M., Izquierdo, L. R., Izquierdo, S. S., Santos, J. I., Olmo, Rd., & López-Paredes, A. (2017). Checking simulations: Detecting and avoiding errors and artefacts. doi: https://doi.org/10.1007/978-3-319-66948-9_9.

Gilbert, N. (2008). *Agent-based models*. Thousand Oaks, CA: SAGE. google-Books-ID: Z3cp0ZBK9UsC.

Grimm, V., Polhill, G., & Touza, J. (2017). Documenting social simulation models: The ODD protocol as a standard. doi: https://doi.org/10.1007/978-3-319-66948-9_10.

Gross, D., & Strand, R. (2000). Can agent-based models assist decisions on large-scale practical problems? A philosophical analysis. *Complexity, 5*(6), 26–33. doi:10.1002/1099-0526(200007/08)5:6<26::AID-CPLX6>3.0.CO;2-G, http://onlinelibrary.wiley.com/doi/10.1002/1099-0526(200007/08)5:6<26::AID-CPLX6>3.0.CO;2-G/abstract

Kratz, J., & Strasser, C. (2014). Data publication consensus and controversies. *F1000Research, 3*, 94. doi:10.12688/f1000research.3979.3, http://f1000research.com/articles/3-94/v3

Laird, J. E. (2012). *The soar cognitive architecture*. Cambridge: MIT Press.

Law, A. M. (2015). Simulation modeling and analysis (5th ed.). New York: McGraw Hill Higher Education.

Lee, J. S., Filatova, T., Ligmann-Zielinska, A., Hassani-Mahmooei, B., Stonedahl, F., Lorscheid, I., et al. (2015). The complexities of agent-based modeling output analysis. *Journal of Artificial Societies and Social Simulation, 18*(4), 4.

McKay, M. D., Beckman, R. J., & Conover, W. J. (1979). Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics, 21*(2), 239–245. doi:10.1080/00401706.1979.10489755, http://dx.doi.org/10.1080/00401706.1979.10489755

Merlone, U., Sonnessa, M., & Terna, P. (2008). Horizontal and vertical multiple implementations in a model of industrial districts. *Journal of Artificial Societies and Social Simulation 11*(2), 5. http://jasss.soc.surrey.ac.uk/11/2/5.html

Miller, J. H. (1998). Active nonlinear tests (ANTs) of complex simulation models. *Management Science, 44*(6), 820–830. doi:10.1287/mnsc.44.6.820, http://pubsonline.informs.org/doi/abs/10.1287/mnsc.44.6.820

Miodownik, D., Cartrite, B., & Bhavnani, R. (2010). Between replication and docking: "adaptive agents, political institutions, and civic traditions" revisited. *Journal of Artificial Societies and Social Simulation, 13*(3), 1.

Müller, B., Bohn, F., Dreßler, G., Groeneveld, J., Klassert, C., Martin, R., et al. (2013), Describing human decisions in agent-based models – ODD + D, an extension of the ODD protocol. *Environmental Modelling & Software, 48*, 37–48. doi:10.1016/j.envsoft.2013.06.003, http://www.sciencedirect.com/science/article/pii/S1364815213001394

Müller, B., Balbi, S., Buchmann, C. M., de Sousa, L., Dressler, G., Groeneveld, J., et al. (2014). Standardised and transparent model descriptions for agent-based models: Current status and prospects. *Environmental Modelling & Software, 55*, 156–163. doi:10.1016/j.envsoft.2014.01.029, http://www.sciencedirect.com/science/article/pii/S1364815214000395

Montgomery, D. C. (2012). *Design and analysis of experiments* (8th ed.). Hoboken: Wiley.

Moss, S., & Edmonds, B. (2005). Sociology and simulation: Statistical and qualitative cross-validation. *American Journal of Sociology, 110*(4), 1095–1131. doi:10.1086/427320, http://www.journals.uchicago.edu/doi/abs/10.1086/427320

North, M. J., Collier, N. T., Ozik, J., Tatara, E. R., Macal, C. M., Bragen, M., et al. (2013). Complex adaptive systems modeling with Repast Simphony. *Complex Adaptive Systems Modeling, 1*(1), 3. doi:10.1186/2194-3206-1-3, http://casmodeling.springeropen.com/articles/10.1186/2194-3206-1-3

Peng, R. D. (2011). Reproducible research in computational science. *Science, 334*(6060), 1226–1227. doi:10.1126/science.1213847, http://science.sciencemag.org/content/334/6060/1226

Pereda, M., Santos, J. I., & Galan, J. M. (2015). *A brief introduction to the use of machine learning techniques in the analysis of agent-based models*. SSRN Scholarly Paper ID 2689676. Rochester, NY: Social Science Research Network. https://papers.ssrn.com/abstract=2689676

R Core Team. (2017). R: A language and environment for statistical computing. https://www.R-project.org/

Radax, W., & Rengs, B. (2009). Prospects and pitfalls of statistical testing: Insights from replicating the demographic prisoner's dilemma. *Journal of Artificial Societies and Social Simulation, 13*(4), 1.

Rollins, N. D., Barton, C. M., Bergin, S., Janssen, M. A., & Lee, A. (2014). A Computational Model Library for publishing model documentation and code. *Environmental Modelling & Software, 61*, 59–64. doi:10.1016/j.envsoft.2014.06.022, http://www.sciencedirect.com/science/article/pii/S1364815214001959

Rouchier, J., Cioffi-Revilla, C., Polhill, J. G., & Takadama, K. (2008). Progress in model-to-model analysis. *Journal of Artificial Societies and Social Simulation, 11*(2), 8. http://jasss.soc.surrey.ac.uk/11/2/8.html

Sargent, R. G. (2013). Verification and validation of simulation models. *Journal of Simulation, 7*(1), 12–24. doi:10.1057/jos.2012.20, https://link.springer.com/article/10.1057/jos.2012.20

Schelling, T. C. (1971). Dynamic models of segregation. *The Journal of Mathematical Sociology, 1*(2), 143–186. doi:10.1080/0022250X.1971.9989794, http://dx.doi.org/10.1080/0022250X.1971.9989794

Squazzoni, F. (Ed.). (2009). *Epistemological aspects of computer simulation in the social sciences*. Lecture notes in computer science (Vol. 5466). Berlin/Heidelberg: Springer. doi:10.1007/978-3-642-01109-2, http://link.springer.com/10.1007/978-3-642-01109-2

Stonedahl, F., & Wilensky, U. (2010). Finding forms of flocking: Evolutionary search in ABM parameter-spaces. In *Multi-agent-based simulation XI* (pp. 61–75). Berlin/Heidelberg: Springer. doi:10.1007/978-3-642-18345-4_5, https://link.springer.com/chapter/10.1007/978-3-642-18345-4_5

Sun, Z., Lorscheid, I., Millington, J. D., Lauf, S., Magliocca, N. R., Groeneveld, J., et al. (2016) Simple or complicated agent-based models? A complicated issue. *Environmental Modelling & Software, 86*, 56–67. doi:10.1016/j.envsoft.2016.09.006, http://www.sciencedirect.com/science/article/pii/S1364815216306041

Takadama, K., Suematsu, Y. L., Sugimoto, N., Nawa, N. E., & Shimohara, K. (2003). Cross-element validation in multiagent-based simulation: Switching learning mechanisms in agents. *Journal of Artificial Societies and Social Simulation, 6*(4), 6. http://jasss.soc.surrey.ac.uk/6/4/6.html

Thiele, J. C., & Grimm, V. (2015). Replicating and breaking models: Good for you and good for ecology. *Oikos, 124*(6), 691–696. doi:10.1111/oik.02170, http://onlinelibrary.wiley.com/doi/10.1111/oik.02170/abstract

Troitzsch, K. G. (2004). Validating simulation models. In G. Horton (Ed.), *Proceedings of 18th European Simulation Multiconference, ESM 2004* (pp. 265–270). Magdeburg: SCS Publishing House.

Wiersma, W. (2015). AgentBase: Agent based modelling in the browser. http://wybowiersma.net/pub/papers/Wiersma,Wybo,AgentBase_agent_based_modelling_in_the_browser.pdf

Wilensky, U. (1999). NetLogo. http://ccl.northwestern.edu/netlogo/

Wilensky, U., & Rand, W. (2007). Making models match: Replicating an agent-based model. *Journal of Artificial Societies and Social Simulation, 10*(4), 2. http://jasss.soc.surrey.ac.uk/10/4/2.html

Will, O., & Hegselmann, R. (2008). A replication that failed – On the computational model in 'Michael W. Macy and Yoshimichi Sato: Trust, cooperation and market formation in the U.S. and Japan. Proceedings of the national academy of sciences, May 2002'. *Journal of Artificial Societies and Social Simulation, 11*(3), 3. http://jasss.soc.surrey.ac.uk/11/3/3.html