

RESEARCH ARTICLE

The Theory-Practice Gap in the Evaluation of Agent-Based Social Simulations

David Anzola

Innovation Center, School of Management, Universidad del Rosario, Bogotá, Colombia
Email: david.anzola@urosario.edu.co

Argument

Agent-based social simulations have historically been evaluated using two criteria: verification and validation. This article questions the adequacy of this dual evaluation scheme. It claims that the scheme does not conform to everyday practices of evaluation, and has, over time, fostered a theory-practice gap in the assessment of social simulations. This gap originates because the dual evaluation scheme, inherited from computer science and software engineering, on one hand, overemphasizes the technical and formal aspects of the implementation process and, on the other hand, misrepresents the connection between the conceptual and the computational model. The mismatch between evaluation theory and practice, it is suggested, might be overcome if practitioners of agent-based social simulation adopt a single criterion evaluation scheme in which: i) the technical/formal issues of the implementation process are tackled as a matter of debugging or instrument calibration, and ii) the epistemological issues surrounding the connection between conceptual and computational models are addressed as a matter of validation.

Keywords: Verification; Validation; Representation; Conceptual modeling; Agent-based modeling; Social simulation

Introduction

Verification and validation are the two most commonly used tests of a computational model's adequacy. In agent-based computational social science, the procedures are generally understood to follow traditional definitions in computer science and software engineering: "Model verification deals with building the model *right* . . . [whereas] model validation deals with building the *right* model" (Balci 1997, 133, emphasis in the original). Though the question of how "right" is to be evaluated for each case is not a matter of general consensus, regardless of the approach, the process of evaluating the adequacy of a simulation can be succinctly depicted according to the flowchart presented in figure 1. Verification focuses on a transformational relationship between the conceptual model (a pre-computational model formulated using natural language, mathematics, pseudocode, visual and graphical aids, etc., which results from the processes of abstracting and concretizing from the phenomenon of interest) and the computational model (a model written in a programming language that is compiled/interpreted by a physical machine), whereas validation focuses on the representational relationship between the computational model and the target phenomenon.¹

In this paper, we argue that this dual evaluation scheme is problematic, for it does not conform to what modelers actually do in everyday practices of evaluation, particularly in regard to the connection between the conceptual and the computational model. We suggest that this scheme

¹Depictions of the process in the literature (e.g. David 2009; Sargent 2013) are often more elaborate, in order, for example, to represent the iterative character of the evaluation. They all, however, take the connection between the conceptual and the computational model to be a matter of verification.

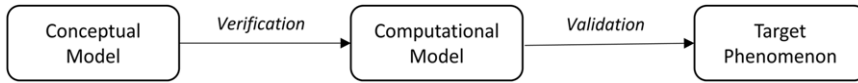


Figure 1. Basic conceptualization of verification and validation.

does not adequately capture the nature of the knowledge produced by agent-based social simulations, and that, by renouncing it, agent-based computational social science could bridge a long-standing theory-practice gap in evaluation practices and set the foundations for a more fruitful exploration of the challenges associated with the evaluation of an agent-based model in the social domain.² This renouncing entails adopting a single criterion evaluation scheme in which i) the technical/formal issues of implementation are tackled as a matter of debugging or instrument calibration, and ii) the epistemological issues surrounding the connection between conceptual and computational models are addressed as a matter of validation.

To illustrate the inadequacy of the dual evaluation scheme, the discussion is structured as follows: the next section approaches the verification-validation distinction as a particular instance of the wider problem of justification of knowledge. It enquires about the nature of the knowledge produced by computer simulation and questions the reasons for having two different criteria in agent-based computational social science. Later, sections three and four show that the current conceptualization of the process of evaluation does not conform to everyday practices of evaluation, first, because it poorly accounts for key representational and experimental features of agent-based social simulation and, second, because it misrepresents some epistemological issues that emerge during the process of implementation. The article closes with a brief discussion about the prospective advantages of adopting a single criterion evaluation scheme.

Verification, validation and the justification of knowledge

In agent-based computational social science, verification and validation practices seek to provide warrants for belief in the adequacy of a computational model. This circumscribes the processes of evaluation within the general problem of knowledge justification. Historically, there have been two major approximations to knowledge testing, depending on whether evidence is produced deductively or inductively. In deductive inference, justified knowledge is linked to the satisfaction of two different criteria: truth and validity. The notion of validity is a formal characteristic of arguments or inferences, while truth is a factual characteristic of statements. The two criteria are evaluated independently and do not interfere with each other. False-valid, false-invalid and true-invalid are three distinct forms of erroneous reasoning and knowledge claims. When deductive inference is both true and valid, warrants for adequacy and justification are powerful. Valid deductive reasoning is truth-preserving (i.e., the conclusions cannot be false if the basic premises are all true). Hence, for a valid inference, only the truth-value of basic premises needs to be tested.

The justification of knowledge claims produced by inductive inference operates in an entirely different way. It depends on finding answers for the problems of induction and perception. The former deals with issues about warrants for belief in knowledge acquired through experience; the latter, with the nature and effects of sense perception. The problem of perception is rarely addressed in the literature on verification and validation in agent-based modeling. This is likely due to the fact that, since modeling is an indirect approach to knowledge, perception is

²While the dual evaluation scheme is transversal to simulation studies, there are key methodological differences that prevent from generalizing the discussion advanced in this text to other disciplinary areas and types of computer simulation. In traditional computer applications, for example, representation is not an issue during the process of verification, given that the final purpose of these applications is functional rather than representational. Likewise, unlike agent-based social simulation, equation-based modeling has to deal with the problem of discretization during the implementation process (Anzola 2021a).

subordinate to representation. Even though the problem of perception could significantly impact warrants for belief in the adequacy of a simulation, it will not be dealt with here, given the historical neglect of perception in the traditional conceptualization of verification and validation in simulation studies.

The problem of induction, conversely, is regularly acknowledged in the agent-based social simulation literature. The standard interpretation of the problem in the philosophy of science assumes that the testing of inductive reasoning is challenging, since it does not have the level of necessitation displayed by deductive reasoning, that is, it is not truth-preserving.³ Induction requires extrapolating from observed instances, using principles and methods of generalization, such as the uniformity of nature (i.e., that there is uniformity between observed and unobserved instances of a phenomenon) and relative frequency (i.e., the higher the number of positive observations, the more likely the inference is true). Because of the reliance on extrapolation from observations, inductive knowledge claims might be erroneous, even when following a proper reasoning, due to the amount, type, and quality of the observations used.

Discussing the full implications of the claim that inductive reasoning is not truth-preserving goes beyond the scope of this text. There is, however, one implication that is worth mentioning: the justification of knowledge in inductive research is often put forward as a matter of validation. There is no dual evaluation scheme because validation cannot be taken entirely as a formal or empirical process. It is the combination of the application of a method and the rational discussion about this application that provides the foundations for justified true belief. Thus, a question arises regarding why knowledge claims produced with computer simulation, which are decidedly inductive, are evaluated using two criteria.⁴

Why two criteria in agent-based computational social science?

The verification-validation distinction was introduced during the seventies in software engineering, and later popularized in general computer science. It was devised as an alternative for the systematization of several quality assurance procedures (Evans 1984). When computational science emerged a decade later, it incorporated the verification-validation evaluation scheme practically unchanged, in spite of the decidedly empirical character of this new area of research.

Given the theoretical and practical diversification experienced by computer science in the last few decades, the difference between computer and computational science might seem superficial. Yet, the distinction is relevant when discussing the dual evaluation scheme. The verification-validation distinction originated in a context in which computer science was believed to center on automatization. Computation was equated to programming and everyday practices of computer science revolved around algorithm analysis. The external world, and the methods for approaching this world, were not disciplinarily relevant at the time (Denning 2010; Tedre 2015). Whereas computer science was focused on the study of computation from a scientific point of view, computational science was interested in using the data-processing power of physical computers to support scientific inquiry (Wolfram 1984; Shiflet and Shiflet 2014). As such, real phenomena, not algorithms or problem-solving applications, became the objects of interest for

³In some contexts, a distinction is made between induction and abduction. That distinction, however, is not necessary if induction is broadly defined as a type of reasoning that is not truth-preserving (and not just inference grounded on statistical probability). We will not elaborate on the distinction here, since for our argument it is enough to highlight that abduction and induction, however they are defined, are both types of reasoning where there is no necessitation and where the evaluation of knowledge claims does not rely on a dual evaluation scheme.

⁴While a computer simulation operates algorithmically, and it could be claimed, following Epstein (1999), that, in social simulation, “from a technical standpoint, *generative implies deductive*” (44, emphasis in the original), knowledge claims generated by operating computational models are not deductive. The execution is approached experimentally (i.e., the model is run multiple times with different parameter configurations) and the validation is regularly carried out using inductive qualitative and quantitative techniques, in evaluation practices that often require calibrating and contrasting the simulation output against external data.

the different branches of computational science, including agent-based computational social science.

The verification-validation distinction was not meant to accommodate or account for knowledge claims i) about the external world and ii) produced by indirect approaches to knowledge (including modeling). The turn to the empirical did not lead to the challenging of the dual evaluation scheme in computer science, since its disciplinary status was partly built upon the assumption that programming is an abstract formal scientific practice (Colburn 2004; Fetzer 2001). Because of this belief in the formal nature of programming, the dual evaluation scheme has been (re)interpreted in empirical research as a method for the assessment of a type of reasoning that is both, or halfway between, deductive and inductive (e.g., Axelrod 1997; Epstein 1999).

The two criteria, verification and validation, allegedly have clear and distinct scopes. The former is meant to focus on formal aspects; the latter, on factual aspects. Verification has an important epistemological status in the practice of agent-based modeling, and computer simulation in general, because it is, to a certain extent, equivalent to validity in the classical sense (i.e., an activity that guarantees the formal adequacy of the algorithmic procedure/deduction). At the same time, the alleged formal nature of verification has led practitioners to strongly emphasize the algorithmic and deductive features of computation. As a result, the existence of two different criteria for the justification of knowledge in agent-based computational social science has somewhat preserved this idea from classical computer science that, if the model is debugged and the crucial elements of the conceptual model are transformed accurately enough into their computational counterparts, the input-output connection in a simulation should have a level of necessitation resembling that of axiomatic systems (Anzola 2019).

The following two sections show that the belief on the alleged formal focus of verification (and, overall, the formal-factual separation in the verification and validation dichotomy), consequence of using the dual evaluation scheme in empirical research, does not conform to everyday practices of evaluation in agent-based computational social science, and has, over the years, fostered a theory-practice gap in the evaluation of knowledge claims. These sections argue that the dual evaluation scheme, on one hand, conflates the technical and epistemological aspects of evaluating a model's adequacy and, on the other hand, misrepresents the nature and role of the conceptual model in social simulation.

Conflating the technical and the epistemological

Definitions of verification in agent-based computational social science are relatively orthodox, reflecting typical concerns about implementation. The process is conceptualized as an evaluation of whether a computational model conforms to the intention of the modeler (Gilbert and Troitzsch 2005; Edmonds 2000; Rand and Rust 2011; Squazzoni 2012; David 2013). Two elements are relatively generalized across definitions: the first is that the degree of conformity to the modeler's intention should be evaluated by checking the correspondence between the computational and the conceptual model. The second is that the evaluation of correspondence is carried out mostly through technical activities, such as debugging. Although definitions of the process of verification regularly include the notion of intentionality, overall, verification is understood, following the traditional conceptualization in computer science and software engineering, mostly as a matter of formal correctness. This emphasis on the formal character of the process makes sense within the context in which the verification-validation distinction originated. Yet, in agent-based social simulation, everyday practices of evaluation involve more than a test of the formal correctness of the implementation.

The effect of empirical content

It has been previously argued, both in computer and computational science, that the definition of verification does not entirely conform to everyday practices (e.g., Anzola 2019; Augusiak, Van den Brink, and Grimm 2014; Graebner 2018; David, Simão, and Coelho 2005, 2007; David 2009; Fetzer 2001; Schulze et al. 2017; Tedre 2015; Winsberg 2010). The discussion so far has focused on characterizing the empirical content of verification and on putting forward some proposals to account for it, without really questioning whether this acknowledgement has any implications for the tenability of the dual evaluation scheme. Perhaps the best way to understand what it means to recognize that verification is empirically permeated is by comparing the process of evaluation in agent-based computational social science with that of typical deductive and inductive approaches in contemporary science.

To begin with, having two separate criteria for the justification of knowledge in agent-based computational social science is problematic because the verification-validation distinction cannot be entirely mapped into the truth-validity distinction of deductive reasoning. Even if verification is taken to center on validity, there would be a problem with the scheme, since validation does not really have an analogue. Validation focuses on an *ex post* contrast with the phenomenon of interest. This *ex post* process is not compatible with the notion of truth preservation of deductive inference, since it would have to either render truth partial or provide information that does not constitute new knowledge. A heavy burden is put on the concept of verification: it would need to account for several representational aspects that go beyond its alleged formal role. If this happens, the distinction between verification and validation gets blurred in a way that is not common in deductive reasoning.

A different sort of problem arises when the comparison is made with inductive methods. Verification, for example, is often called “internal validation” in agent-based computational social science and general computer science (Axelrod 1997; Cioffi-Revilla 2014; Galán et al. 2009; Gilbert and Troitzsch 2005). The concept of internal validation comes from the experimental/quasi-experimental domain. It is basically an inquiry about whether there is an identifiable causal connection in the manipulation of the experiment (Campbell 1957). Because of its emphasis on inference, the implications of internal validation in experimental settings are different to those of verification in agent-based modeling. Regardless of whether it is internal or external, in experimental settings, “validity is not a property of methods but of inferences and knowledge claims” (Shadish, Cook, and Campbell 2002, 480). Verification, if conceptualized as centering on the formal correctness of implementation, should not be taken to imply any form of inference or knowledge claim about the phenomenon of interest in social simulation. This is due, in part, to the epistemological nature of evaluation schemes with two criteria. In deductive inference, validity, the criterion focused on formal correctness, does not inquire about truth-values, so it cannot account for any factual content associated with the inference. Without empirical content, knowledge claims about the external world cannot be produced. There is, then, a major epistemological difference between verification and internal validation that is being overlooked when the two concepts are used interchangeably.

An additional downside of conceptualizing verification as an instance of internal validation is that it misrepresents the connection between internal and external validation. In the experimental/quasi-experimental domain, there is a question linking the two forms of validity: whether the *ceteris paribus* conditions of the experimental manipulation could be identified in the target phenomenon (i.e., whether the experimental results are projectable or generalizable). In agent-based computational social science, conversely, there is no clear connection between verification and validation. Definitions regularly emphasize the difference, not the connection, between the two concepts. It is possible the link between the two is rarely discussed because of the belief that verification operates in the formal domain, whereas validation does it in the empirical. As a result, the concerns addressed by the internal-external validity distinction in experimental settings cannot be mapped into the verification-validation distinction in agent-based computational social

science. If anything, they should be exclusively associated with the concept of validation. A contradictory situation ensues: when agent-based modeling is compared with deductive inference, verification seems to account for the two criteria of evaluation; when the comparison is with experiments, it is validation that accounts for the two forms of validity.

The verification-validation and internal-external validity dualisms do not overlap because of the empirical character of verification, but also because agent-based modeling is not just a form of experimentation. The method is described in the literature as a type of formal theory, experimentation, modeling, or any combination thereof. This ambiguity regarding the methodological nature of agent-based modeling impacts the process of evaluation, since criteria for the justification of knowledge are developed differently for each method. In experimental settings, for example, knowledge adequacy is often linked to concerns about parallelism. The assessment of parallelism centers on the context of interventions and the *ceteris paribus* conditions that underpin the experimental manipulation (Guala 2002; Morgan 2003). In modeling, conversely, knowledge adequacy is mostly linked to concerns about representation. Representation, in comparison to parallelism, uses criteria to test knowledge claims that are more flexible and are usually linked to the physical, cognitive, and social aspects of the context of modeling (Anzola 2021a, Frigg and Nguyen 2017).

Verification and representation

Verification is not only problematic when agent-based social simulation is considered a type of experiment, but also when it is considered a type of modeling. As it is currently conceptualized, verification downplays issues of representation. In the agent-based and general simulation literature, it is often claimed that “in verification, the association or relationship of the simulation to the real world is not an issue” (Müller and von Storch 2004, 13) and that “verification has more to do with debugging and software engineering than it does with a comparison of the model to the real-world” (Louie and Carley 2008, 245). Contrary to this widespread belief, the connection between the conceptual and the computational model cannot be made sense of without including judgments about representation. Naturally, however, not every verification technique incorporates representational concerns in the same way or to the same extent. Take, for example, “corner testing” and “running the simulation with known parameter values”—two well-known verification techniques (Gilbert 2008; Rand and Rust 2011). For the first one, the simulation is run with extreme or unrealistic parameters (e.g., no agents) to test for some evident errors. If a simulation with no agents produces some output associated with interaction, technical aspects of the model are checked (e.g., whether all the information from previous runs is cleared before a new run starts). The conceptual model is unlikely to be revised, for corner testing deals with parameter configurations that are not likely to find real world correlates (at least for plausible or relevant configurations of the model). Even though there is an external referent for the phenomenon being modeled, representational issues about the target phenomenon can be solved in a straightforward manner.

The situation is different when verifying the model using known parameter values. An output mismatch identified by the application of this technique is corrected either by calibrating the parameters or by revising the implementation. While a purely technical issue might be the cause of the mismatch, reproducing known scenarios and corner testing posit different epistemological challenges. The former has to deal with the risk of overfitting a model in a way the latter does not. To fully capture the implications of this claim, the conceptualization of the evaluation process needs to be revisited to introduce a third model, the “model of the phenomenon”: a post-computational conceptual model that seeks to make sense of the simulation results through a narrative that links back to both the computational and the conceptual model (figure 2). The model of the phenomenon is the most elaborate form of representation in the simulation life cycle. It is used, initially, to make sense of emergent dynamics during the early stages of

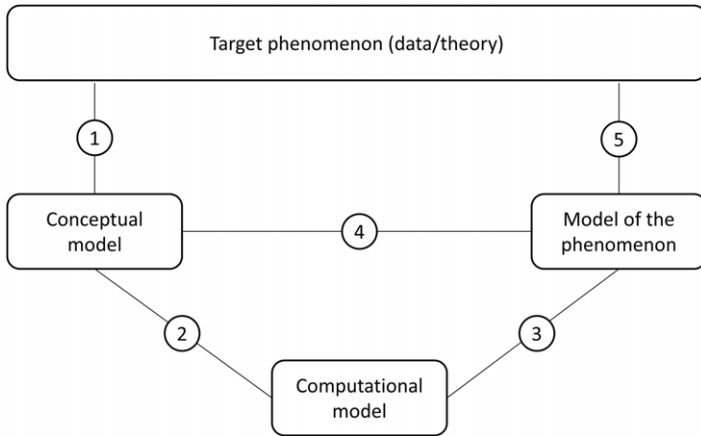


Figure 2. Alternative conceptualization of the evaluation process. It (i) introduces the “model of the phenomenon,” (ii) makes explicit the connection between the conceptual model and the model of the phenomenon, (iii) explicitly connects the model of the phenomenon and the target phenomenon (reconstructed as data and/or theory), and, finally, (iv) depicts the connection between the three models as an iterative process.

the simulation; later, it helps to test knowledge claims, following a comparison with the target phenomenon.

Neither corner testing nor reproducing known scenarios allows for the formulation of robust (new) knowledge claims about the phenomenon of interest. Yet, they both deal with the way this phenomenon is represented, in both the conceptual and the computational model. This is why the model of the phenomenon needs to be introduced in the scheme. Practitioners often talk as if there was a direct comparison between the computational model and the target phenomenon. In addition, graphical representations of the process of evaluation, both in computer science and agent-based computational social science (e.g., Louie and Carley 2008; Sargent 2013), depict the process this way. Representational issues, however, are regularly addressed during early stages of the simulation process through sense-making processes that do not involve a comparison with the phenomenon of interest as such. The model of the phenomenon, as a narrative that allows for increasing sense-making of the execution of the computational model, is used to deal with these representational issues before the simulation output is considered to have provided enough warrants for belief in the adequacy of the results.

Corner testing is a technique that basically addresses relationship number (2) described in figure 2. Using known parameter values, on the contrary, also involves relationships (3) and (4). The researcher needs at least a basic idea of the structural and functional properties of the computational model and the target phenomenon in order to judge on adequacy, for it is not immediately clear whether unexpected results originate from the articulation of the conceptual model, the transition from the conceptual to the computational model or the execution of the simulation. Decisions about adequacy could lead to further modifications of the conceptual or the computational model. They could also lead to challenging the adequacy and accuracy of background data that the model is fitted with or contrasted against.

Since running the simulation with known parameter values deals with real-life comparisons, criteria to decide whether there is a mismatch between the output and the available data need to be developed. These criteria depend on several representational issues regarding the objects being represented, their properties and the way these properties are measured. This is a crucial element for agent-based modeling in social science, since results are not analytically derived and there could be multiple (non-equivalent) implementations of the same phenomenon (Anzola 2021c). Because there is not always an output that is readily available for contrast with data

produced by previous studies and models, a “domain of validity” (Axtell et al. 1996) must be established prior to the implementation of verification techniques, such as using known parameter values. In addition, the bottom-up and numerical nature of agent-based models create a larger surface response, which requires a revision of the criteria on which goodness-of-fit is usually judged.

The temporal asymmetry in verification-validation

The formal or technical adequacy of an agent-based model cannot be guaranteed by the application of different techniques in the early stages of the simulation life cycle. Not enough warrants for belief in the adequacy of the model are produced by techniques in which representation is not yet a concern. However, since representation becomes an issue from very early on, it could be questioned whether there is a solid foundation for the verification-validation distinction. The goals of both processes basically overlap through the entire the simulation life cycle. Confidence in the output of the computational model increases with the reiterative application of different evaluation techniques over time. While some techniques are better suited for early stages of the evaluation process, the division established by the dual evaluation scheme might be both erroneous and unnecessary.

The reason why corner testing or running the simulation with known parameter values are used early during the simulation life cycle is because they do not require an intricate and robust model of the phenomenon, not because they do not deal with representational issues at all. Any evaluation technique sets particular representational requirements for the model of the phenomenon that can naturally be ordered in an interval, according to their impact on the evaluation process. Those at the lower end of the spectrum, where representation is not much of a concern, can still have an effect or be employed in later stages of the simulation life cycle. Understanding about these different representational requirements, however, is hindered by the way the verification-validation distinction is currently depicted. The dual evaluations scheme only has a very clear and distinct ambit of application when it is limited to technical concerns, such as making sure the simulation does not crash. Beyond that, a conflation of the technical and the epistemological, particularly the representational, occurs.

In the practice of modeling, technical and representational issues are often entwined, for the process of modeling and the model itself, not just the results, constitute the source of inference and knowledge claims (Frigg and Nguyen 2017; Knuuttila 2021). Warrants for belief in the adequacy of a computational model are produced by a progressive increase in confidence in the simulation, both as an object of knowledge and as an instrument. It is this two-sided increase in confidence that determines, as is common with inductive methods, when it is suitable to move from calibration to experimentation and when the simulation results provide enough warrants for belief to end the experimental phase (Arabatzis 2013; Galison 1987). Although the simulation as such has an algorithmic behavior with a clear beginning and end, the evaluation of a simulation’s adequacy is constrained by the same time-horizon uncertainty of empirical inductive inference.

The current verification-validation distinction in agent-based computational social science, however, is ill-suited to account for representational and experimental concerns in the way just described. Typical definitions incorrectly displace representational issues to the validation process. In addition, by failing to distinguish between the purely technical and the epistemological aspects of verification, agent-based computational social science has articulated an account of evaluation that misrepresents key features of other empirical approaches in contemporary science. To call verification “internal validation,” for example, not only creates confusion about the goals and features of the process, but also mistakenly leads practitioners to overemphasize the algorithmic aspects of the method, while downplaying the experimental and representational.

The transition from the conceptual to the computational model

The conceptualization of the process of implementation, on which verification itself relies, is also problematic in the context of agent-based computational social science. In computer science and software engineering, implementation has been dealt with mostly as the translation of algorithms into computer code. As such, the connection between the conceptual and computational model is often depicted as linguistic transformation. Verification, however, could only be considered a formal process of translation if the connection between the conceptual and the computational model is oversimplified and the character of the conceptual model is taken as unproblematic. Everyday practices provide evidence to suggest that neither assumption is correct in agent-based social simulation.

The lack of knowledge about conceptual models

The conceptual model is a fundamental element of any simulation. It is the result of the initial abstraction process associated with the domain, scope and level of detail of the simulation. It has, therefore, a significant influence on the technical and epistemological goals set for the simulation. The conceptual model also determines the roadmap for the different stages of the simulation life cycle and provides the bases for communication, as well as means for control and evaluation (Balci 2003; Robinson 2011). Conceptual models, in practice, can range from very general descriptions of the simulation goals (along with some technical aspects, such as input, output and mechanisms), to very detailed pseudocode that stands almost in a one-to-one relationship with the implemented computer code (Robinson 2011; Wang and Brooks 2011; Willemain 1994).

Because of the diversity in the approach to conceptual modeling in computer simulation, the process is often considered more as a craft (Robinson 2011; van der Zee et al. 2011). The “craft” label is not meant to imply the process is entirely subjective, but, rather, that it is heavily context-dependent. In specialized literature, the term “craft” is usually used when referring to a process of making that involves particular domain-specific skills, a sense of mastery and a sense of uniqueness (Cardoso 2010; Shiner 2012). The practice of modeling depends on several features of the modeler, the target audience and the context of modeling, such as skills, goals, background knowledge, computer power, interests and values. In agent-based computational social science, the conceptual model, as well as the subsequent computational model, are the result, first, of the way in which material, social and cognitive resources are accessed and mobilized and, second, of the principles and assumptions surrounding the practice of agent-based modeling (e.g., the time-evolution of the simulation accounts for emergence). Neither resources nor assumptions are uniform and consistent enough throughout the field to assume that the connection between the conceptual and the computational model can be expressed formally, without reference to the context in which the model is developed.

The belief that implementation is an activity of a linguistic nature has persisted in an applied field such as agent-based computational social science for several reasons. One such reason is that scientific accounts of conceptual modeling are scarce and not standardized (Robinson 2020). Second, most accounts are developed after other types of simulation (mostly discrete-event simulation) and in relatively narrow research areas (operational or land research) (Argent et al. 2016; Brooks and Wang 2015; Powell and Willemain 2007). Third, issues pertaining to the process of conceptual modeling are not reported in academic publications, so there has not been much incentive for practitioners to reflect on how they approach this practice.

Even though new reporting tools such as protocols encourage modelers to make the process of conceptual modelling more explicit, most knowledge about it remains tacit. There is no information, for example, regarding whether formal or natural languages are more commonly used or whether conceptual models are mostly linguistic, ideogrammic, material or a particular combination thereof. There is no knowledge either about how different principles and tensions

(e.g., the KISS-KIDS debate)⁵ inform the process of conceptual modeling. The lack of research on everyday practices of implementation directly impacts the evaluation of agent-based social simulations, since it hinders the identification of those aspects of the implementation that go beyond the coding of the computational model.

The conceptual-computational transition as translation

Even if conceptual models in the field were mostly linguistic, the implementation could hardly be taken as a matter of formal translation or mapping, since the transition between the conceptual and the computational model does not follow a predefined set of formalization rules. Formal deductive systems use predetermined languages, in which an exhaustive description of objects and relations is provided. It is these formalization rules that allow for the process to be thought of as mapping or translation. The sentence “All men are mortal,” for example, is formalized in predicate logic as $\forall x(man(x) \Rightarrow mortal(x))$. The determiner and verb are replaced by symbols expressing logical relations, following clear predefined transformation rules. The formalization could be taken further by also replacing the empirical content associated with the descriptive terms “man” and “mortal”—for example, $\forall x(ma(x) \Rightarrow mo(x))$. Such transformation, however, would require accounting for “Ma” and “Mo” through additional transformation rules.

Computer languages are certainly a type of formal language, in which the vagueness and ambiguity of natural languages is avoided. However, the process of computational modeling is so underdetermined that there is no guarantee a sentence of the type “All men are mortal” will always yield a formalized predicate of the type $\forall x(man(x) \Rightarrow mortal(x))$. The first reason is technical. The same algorithm can be coded, compiled and executed in diverse ways, depending on the features of the language, software and hardware. Variations on these three aspects could make the simulation yield different results that could eventually affect warrants for belief in the adequacy of a simulation. The second reason is contextual. Depending on the type of model, the transition from conceptual to computational model could imply either a reduction or increase in linguistic complexity or computational expressiveness.⁶ If the conceptual model is formulated in natural language, the computational model would likely imply a reduction; if it is formulated in mathematics, an increase. A computational spatially explicit iterated game, for instance, is more complex and expressive than the mathematical formulation of the game, but is less complex and expressive than the formulation of a social dilemma in natural language. Given that conceptual models often combine several languages, most implementation processes would require entwined processes of increase and reduction of linguistic complexity and computational expressiveness.

Along with the fact that agent-based computational social science has not produced any set of predefined formalization rules able to guide the process of implementation, and that sentences to be translated would be significantly more complex than “All men are mortal,” there are limitations to the potential success of those rule sets in making the process one of formal mapping or translation. Problems of implementation in agent-based computational social science have been previously linked to the non-formal nature of most social theory (David, Simão, and Coelho 2005; 2007). Yet, regardless of the level of formalization, translation is always subject to what, in philosophy of science, is known as indeterminacy of translation (Quine 2013). As originally formulated by Quine, this concept has far-reaching philosophical implications that led, for example, to the

⁵KISS-KIDS (Keep it Simple, Stupid-Keep it Descriptive, Stupid) is a debate about how complicated a model should be to provide reliable knowledge about the target phenomena. Initially, practitioners of agent-based social simulation displayed a preference for simple, abstract models, claiming this modeling style made it possible to identify underlying mechanisms more easily. Over time, however, there has been a progressive popularization of intricate, empirically calibrated models, partly because of an increasing interest in using these models for prediction and control (Anzola 2021b).

⁶Both “linguistic complexity” and “computational expressiveness” address diversity in realization of meanings, symbols and structures, and their connections in a language. The former originated and is commonly used in linguistics; the latter, in computer science.

popularization of naturalized epistemology. For the present context, the idea of indeterminacy of translation is relevant because it suggests that there is no set of formalization/translation rules that could render the translation between any two languages univocal. Hence, the coding of a computational model is an empirical activity that will always require contrast with both background knowledge about the phenomenon of interest and prior assumptions about the operation of the model.

The idea, then, that the relationship between the conceptual and the computational model is one of translation is only tenable if the notion of translation is used in a very loose sense. That is, in the sense that the computational model reflects in a formal way, because it is formulated in a programming language, concerns that have been previously identified during the formulation of the conceptual model. The process, however, is far from being formally determined. The high diversity in implementation practices ensures that no exhaustive and unambiguous criteria for the transition from one model to the other could be devised. Although the simulation operates algorithmically, not all the empirical content of the conceptual model is accounted for in the computational model. In part, that is why, in spite of the formal character of programming languages, practitioners face noticeable difficulties during processes of replication and model-to-model comparison.

The reason one might use a conceptual model to evaluate a simulation is not that there exists the possibility of a formal syntactic and semantic transformation into the computational model. It is, instead, that, in everyday practices of simulation, the model serves as an external reference that is relatively simple and easily accessible. This relationship has yet to be addressed in the evaluation literature because, in most instances of modeling, it is accessed informally and unsystematically (Brooks and Wang 2015; Powell and Willemain 2007). While adopting modeling practices that allow for traceability (e.g., Scherer et al. 2015) could certainly help make explicit the way in which practitioners resort to different models and relationships among these models for sense-making, there is still not a widespread acknowledgement of this need in agent-based social simulation.

The everyday role of the conceptual model

Apart from the underdetermination of the process of conceptual modeling and the evident empirical content in the connection between conceptual and computational model, the alleged formal nature of verification is undermined by everyday practices of agent-based social simulation because the conceptual model still plays a fundamental role once the computational model is implemented. As discussed above, even for a simple technique such as using known parameter values, the relationship between the conceptual model and the computational model cannot be understood only by focusing on connection number (2) in [figure 2](#), but requires understanding about numbers (3) and (4), as well. Representational issues associated with the use of several verification techniques require the articulation of a model of the phenomenon with which both the conceptual and the computational model can be contrasted.

In addition, moving from the conceptual to the computational model is neither a one-off, nor sometimes a one-to-one, process. The implementation of the computational model might involve complex subsequent revisions and modifications. These modifications, however, can go both ways. A mismatch that occurred while trying to reproduce known scenarios, for example, might lead to a modification of the conceptual model, leaving the computational model intact. The conceptual model is not necessarily a precise well-defined model that is only used to inform the implementation of the computational model. The relationship between conceptual and computational model is bidirectional: it is one of syntactic and semantic transformation, as much as one of informal mutual sense-making (Mcnamara et al. 2008; Norling, Edmonds, and Meyer 2013; Pool 2011; Robinson 2011).

The conceptual model also intervenes in processes of informal sense-making with the model of the phenomenon. The connection between the conceptual model and the model of the phenomenon is particularly distinctive in agent-based computational social science, since it is meant to account for processes of emergence. Executing the computational model provides an output that is not entirely known beforehand (i.e., it is not part of the specifications that are identified in the conceptual model and later implemented into the computational model). Since the conceptual model is relatively simple and easily accessible, it helps with testing the plausibility of emergent patterns that are conceptualized in the model of the phenomenon after the simulation is executed, especially in cases where the simulation produces output that is surprising or contradictory, when compared with previous data or the modeler's expectations. This is a role of conceptual models, addressed by connection number (4) in [figure 2](#), that is rarely accounted for, even by conceptualizations of the evaluation process that include the model of the phenomenon (e.g., David 2009).

The process of sense-making sometimes requires different activities, such as implementing surrogate computational models, the inclusion, exclusion or substitution of data, and interaction with stakeholders. Troitzsch (2016), for example, reports the case of a toy model that was initially developed to inform and make sense of the implementation of another more robust model (Nardin et al. 2016), but ended up becoming a fully-fledged computational model on its own. After adding a visualization interface and converting the model from period- to event-oriented, an independent and distinct model of the phenomenon was developed for this model and later compared to the target phenomenon. Accounting for emergence in agent-based social simulation usually leads to successive modifications of the three models after the first iteration of the simulation process (i.e., the generation of a first full set of interconnected models: conceptual, computational and of the phenomenon). At the beginning, the sense-making can be relatively simple, with a distinctive technical component. Yet, as the number of iterations increases, the process of sense-making moves further away from technical concerns and is less likely to be reconstructed in formal terms, since it could involve a diverse set of activities that do not depend exclusively on the implementation and operation of the computational model.

As with the experimental and representational features of agent-based social simulation, the current conceptualization of the verification-validation distinction is both inadequate and insufficient to understand the process of implementation. It is inadequate because this process is not about linguistic mapping or translation. Even if it were, the mapping would still be heavily empirically permeated. It is insufficient because it fails to acknowledge the conceptual and material diversity in the process of conceptual modelling, as well as the impact the conceptual model has on the process of evaluation, even after the implementation of the computational model. The sense-making associated with the connection between the three models cannot be fitted into the traditional formal-empirical or technical-representational division of the verification-validation distinction.

Renouncing the dual evaluation scheme

Given that verification involves several non-technical issues, particularly of representation, and that the connection between the conceptual and the computational model is not just a matter of formal translation, one might well ask whether renouncing the dual evaluation scheme could help close the theory-practice gap in the evaluation of agent-based models in the social domain. In what follows, five reasons are provided in support of a positive answer to this question. From different standpoints, these reasons present evidence that everyday practices of evaluation could be more adequately accounted for within a single criteria evaluation scheme.

A first reason for dispensing with the dual evaluation scheme is that the formal or technical part of the evaluation is so basic that it should not confer upon verification the dominant epistemological status it currently has. All empirical studies go through a phase of technical/formal testing

or design, usually during the early stages of the research process. However, they rarely report on it, since it is often not considered crucial for the knowledge claims advanced. Most studies, for example, do not report on pilot tests and most do not take the data from these tests into consideration for the final analyses and results.

The alleged distinctiveness of verification is hard to justify when compared with common inductive inference and research methods. A more adequate analogue for verification in the empirical domain is instrument calibration, not internal validity, as is usually claimed. Instrument calibration centers, as verification is meant to do, on the increase of confidence in the instrument. It is fundamental, for example, to guarantee the accuracy of the measurement. It is certainly important from a methodological point of view, but, by itself, instrument calibration does not provide any new knowledge claims. It seems likely the same should be said about the role played by the technical part of verification in agent-based computational social science.

This comparison could be taken further. While some might suggest keeping “verification” for technical and formal concerns exclusively, the field would be better off using alternative terms such as “debugging” and “instrument calibration.” These concepts are widespread and defined more precisely in contemporary science. The terminological change would allow practitioners to easily explore methodological similarities and differences between agent-based social simulation and other forms of empirical research. In addition, using clear and standardized concepts would shift attention toward the epistemological aspects of social simulation that are being neglected by the alleged formal focus of verification.

The second reason that would make a single criterion evaluation scheme advantageous is that the separation between formal and representational is not only difficult, but also undesirable. It hinders understanding of agent-based modeling as a particular form of scientific modeling. Goals behind the construction and operation of models rely on the particular material, cognitive and social setting in which the individuals and groups associated with the model, either through design, implementation, operation or communication, are immersed. This is why the connection between the different models partly hinges on processes of sense-making: these processes determine the extent to which a model could be said to represent the world. The KISS-KIDS debate, for example, with its focus on the connection between simplicity and understanding, has significantly influenced modeling choices and epistemic goals in agent-based computational social science (Anzola 2021b). Representational and experimental concerns that are pushed into the background due to the alleged formal nature of verification are fundamental to making sense of the way agent-based models are used to produce knowledge about the world.

In addition, the formal/technical-representational separation is only compatible with a view of simulation as an aid for computation. This longstanding view in equation-based modeling does not fit the practice of agent-based modeling. Given the semantic and syntactic flexibility of social theory and programming languages, agent-based models are often articulated using unique approaches and reconstructions of the theories of properties and measurement in the social domain (Anzola 2021c). As mentioned, material, cognitive and social resources influence the different stages of the simulation life cycle in such a way so as to make the model itself the object of inquiry. When this happens, it is difficult to separate the effect of technical and representational issues on warrants for belief in the adequacy of a computer simulation.

Some might consider alternative conceptualizations of the evaluation process as a possible measure to prevent dropping the dual evaluation scheme. One option, for example, might be to separate verification into more than one category. David, Simão, and Coelho (2005; 2007), for example, divide the notion of verification into formal, empirical and intentional. In their scheme, the formal and technical aspects of verification (e.g., debugging), are covered by the first category. The second category, empirical verification, focuses on implementation, acknowledging it is a process with empirical underpinnings. Finally, intentional verification centers on how epistemological and non-epistemological aspects of the process of evaluation are dealt with.

Even though David *et al.* acknowledge that verification is not just about formal and technical issues, the categories of empirical and intentional verification might pose additional problems. The former relies on the assumption that “insofar as a program can be considered a particular encoding of an algorithm suitable for compilation and execution in a computer, that program may be qualified as a causal model of a logical structure that instantiates a particular algorithm” (David, Simão, and Coelho 2005). This, according to the authors, is a distinctive quality of computational models, not shared by theories or mathematical proofs. That, however, is only true for traditional accounts of scientific theories. In contemporary science, theories are mostly understood in terms of their realization or instantiation in models that need not be linguistic (Knuuttila 2021; van Fraassen 2008). Because of this, models in other empirical fields of research could have the same causal capability of computational models. There would be a need, then, to show that there is something distinctive about the notion of empirical verification that would prevent dealing with this issue, as it is done in other model-based areas of research, using a single criterion of evaluation.

The concept of intentional verification is problematic for a different reason. According to the authors, the role of the concept is to “exercise the construction of specifications and programs in order to achieve experimental adequacy between program executions and the intentional meaning of those programs, always in the context of some limited community of observers” (David, Simão, and Coelho 2005). While the notion of intentional verification evidences the acknowledgement that simulations are not evaluated through a first-order analysis, but through a narrative that, first, makes sense of the operation of the model and, second, is employed for communication and socialization of the results, it delimits the formulation of knowledge claims to the operation of the model. The authors assume that “if computer science is regarded as an empirical science, then the experimental reference of any theory about the computation of a program in an abstract machine consists of executing that program in a target machine” (*ibid.*). By focusing on the operation of the model, the category of intentional verification neglects the fact that several warrants for belief are linked to issues of representation. These warrants are determined by the use of the computational model as a surrogate for thinking, not by the experimental exploration of structural and functional properties of the model. Often, some of these issues are addressed in connections number (1) and (5) in figure 2 and, as such, are established and operate independently of the computational model being evaluated.

The third reason for which practitioners might be better off using a single criterion evaluation scheme is that common definitions of verification tend to involve more than the technical, the experimental and the representational. Gilbert & Troitzsch (2008), for example, suggest that, along with debugging, the process of verification should include better programming practices, such as coding elegantly and adding comments to the code. In turn, Rand and Rust (2011) suggest it should include a sub-stage they call “documentation”: a process of producing extensive description of the conceptual and the computational model, at a degree of specificity that allows for an easy comparison between the two. These additional tasks or subprocesses involve knowledge that is not embedded in the model and does not depend on implementation, but emerges, instead, from beliefs and assumptions about the general practice of science. Beliefs about what constitutes a good model description, for example, might depend on whether it allows other researchers to replicate the model in question. Because of the way the dual evaluation scheme is conceptualized, these criteria that operate independently of the models, mostly through interaction among practitioners, are usually not made explicit and discussed in evaluation practices.

A fourth reason for abandoning the dual evaluation scheme in agent-based computational social science is that it is pragmatic. The frequency with which verification and validation overlap is just too high. Even the distinction between formal, empirical and intentional verification discussed above is undermined by the lack of separation between the concepts. According to David, Simão, and Coelho (2005), “*verification* is to ascertain the validity of certain output as a function of given input, regardless of any interpretation given in terms of any theory or any

phenomenon not strictly computational, . . . [while] *validation* is to ascertain that the execution of a program behaves according to the relatively arbitrary expectations of the program end-users” (emphasis in the original). The reference to intentionality that most authors include in the definition of verification appears in this case, instead, linked to validation. Simplicity and non-redundancy are common sought for qualities in scientific inquiry. Semantic imprecision and redundancy would be avoided if a single criterion evaluation scheme is adopted.

Finally, practitioners might be better off dispensing with the verification-validation distinction to avoid putting themselves in the difficult position of having to justify the need for a dual evaluation scheme. Since the verification-validation distinction does not directly map into the truth-validity dualism of deductive reasoning, practitioners of agent-based social simulation would need to articulate an account of explanation grounded on the assumption that computer simulation constitutes a third type of knowledge that is neither deductive nor inductive, and requires a different dual evaluation scheme. While there is still a domain of the purely formal or technical, a better option might be to approach the justification of knowledge in agent-based modeling as a matter of validation. This would make agent-based computational social science an instance of empirical inductive research, not a distinct type of knowledge source on its own.

Overall, renouncing the dual evaluation scheme could increase awareness about the material, cognitive and social elements that are embedded in the computational model and the practice of modeling. The justification of knowledge produced by agent-based models requires paying attention to the tacit knowledge surrounding the processes of design, implementation, operation and communication. While these aspects are rarely explicitly addressed in evaluation practices and scientific reporting, they are needed to adequately conceptualize the neglected processes number (1) and (5) in [figure 2](#). During (1), for example, a key transformational relationship: between the mental model, the first model in the mind of the researcher, and the conceptual model, takes place. Even though the mental model has not been addressed in the agent-based social simulation evaluation literature, it is important to understand how general assumptions about representation are articulated in simulation practices. While some, for example, have previously argued that the researcher offloads cognitive content to the computational model, others consider that it is more appropriate to address simulation practices as involving a coupling of internal and external representations (Nersessian and MacLeod 2017). It is likely that judgements about representation will be affected by the perceived role of computer simulation in the representational testing of mental models.

Conclusions

This article focused on the verification and validation of agent-based models as a particular instance of the wider philosophical problem of knowledge justification. Unlike mainstream empirical research, agent-based computational social science has maintained two different criteria for the evaluation of a simulation’s adequacy. This dual evaluation scheme, it was argued, is responsible for a theory-practice gap in the process of evaluation. It, on one hand, does not properly account for the representational and experimental elements of the method, since it misrepresents the effect of the empirical nature of social simulation, and, on the other hand, erroneously assumes that implementing the computational model involves a unidirectional relationship of linguistic mapping or translation and that the process of conceptual modeling is well articulated and standardized.

As a part of computational science, a challenge is there for practitioners of agent-based social simulation to develop a method of evaluation that responds to the fact that this area of research centers on target phenomena that are objects and processes in the real world and not just algorithms, as was the case when the verification-validation distinction originated. An alternative depiction of the process of evaluation was presented ([figure 2](#)), in order to start moving in that direction.

In comparison to previous depictions of the evaluation process, [figure 2](#) is evidently underspecified. That was a deliberate decision. There is a noticeable lack of information about the process of modeling in agent-based computational social science, beyond what little is reported in academic publications and what is thought to be true about the field through comparison with other social research methods or other forms of computer simulation. Further research on agent-based social simulation *as a practice* must be carried out to fill the gaps. Conceptual modeling, for example, is often linked in the literature to process number (1). Yet, several instances are found in which the conceptual model is affected both by the implemented computational model and the model of the phenomenon, that is, numbers (2) and (4) in [figure 2](#). Only by studying *actual* practices of modeling will evaluation theory be able to identify and characterize the elements that influence warrants for belief in the adequacy of agent-based models.

Disclosure statement. The Author(s) declare(s) that there is no conflict of interest.

Funding. This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

References

- Anzola, David. 2019. "Knowledge Transfer in Agent-Based Computational Social Science." *Studies in History and Philosophy of Science Part A* 77: 29–38.
- Anzola, David. 2021a. "Capturing the Representational and the Experimental in the Modelling of Artificial Societies." *European Journal for Philosophy of Science* 11. <https://doi.org/10.1007/s13194-021-00382-5>
- Anzola, David. 2021b. "Disagreement in Discipline-Building Processes." *Synthese* 198: 6201–6224.
- Anzola, David. 2021c. "Social Epistemology and Validation in Agent-Based Social Simulation." *Philosophy & Technology* 34: 1333–1361.
- Arabatzis, Theodore. 2013. "Experiment." In *The Routledge Companion to Philosophy of Science*, edited by Stathis Psillos and Martin Curd, 191–202. New York: Routledge.
- Axtell, Robert, Richard Sojda, Carlo Guipponi, Brian McIntosh, Alexey Voinov, and Holger Maier. 2016. "Best Practices for Conceptual Modelling in Environmental Planning and Management." *Environmental Modelling and Software* 80: 113–21.
- Augusiak, Jacqueline, Paul Van den Brink, and Volker Grimm. 2014. "Merging Validation and Evaluation of Ecological Models to 'Evaluation': A Review of Terminology and a Practical Approach." *Ecological Modelling* 280: 117–28.
- Axelrod, Robert. 1997. "Advancing the Art of Simulation in the Social Sciences." In *Simulating Social Phenomena*, edited by Rosaria Conte, Rainer Hegselmann, and Pietro Terna, 21–40. Berlin: Springer.
- Axtell, Robert, Robert Axelrod, Joshua Epstein, and Michael Cohen. 1996. "Aligning Simulation Models: A Case Study and Results." *Computational & Mathematical Organization Theory* 1 (2): 123–41.
- Balci, Osman. 1997. "Verification, Validation and Accreditation of Simulation Models." In *Proceedings of the 1997 Winter Simulation Conference*, edited by Sigrun Andradóttir, Kevin Healy, David Withers, and Barry Nelson, 135–41. Atlanta: IEEE.
- Balci, Osman. 2003. "Verification, Validation, and Certification of Modeling and Simulation Applications." In *Proceedings of the 2003 Winter Simulation Conference*, edited by Stephen Chick, Paul Sánchez, David Ferrin, and Douglas Morrice. New Orleans: IEEE.
- Brooks, Roger, and Wang Wang. 2015. "Conceptual Modelling and the Project Process in Real Simulation Projects: A Survey of Simulation Modellers." *Journal of the Operational Research Society* 66 (10): 1669–85.
- Campbell, Donald. 1957. "Factors Relevant to the Validity of Experiments in Social Settings." *Psychological Bulletin* 54 (4): 297–312.
- Cardoso, Rafael. 2010. "Craft Versus Design: Moving Beyond a Tired Dichotomy." In *The Craft Reader*, edited by Glenn Adamson, 321–332. Oxford: Berg Publishers.
- Cioffi-Revilla, Claudio. 2014. *Introduction to Computational Social Science*. Berlin: Springer.
- Colburn, Timothy. 2004. "Methodology of Computer Science." In *The Blackwell Guide to the Philosophy of Computing and Information*, edited by Luciano Floridi, 318–326. New York: Blackwell.
- David, Nuno. 2009. "Validation and Verification in Social Simulation: Patterns and Clarification of Terminology." In *Epistemological Aspects of Computer Simulation in the Social Sciences*, edited by Flaminio Squazzoni, 117–129. Berlin: Springer.
- David, Nuno. 2013. "Validating Simulations." In *Simulating Social Complexity*, edited by Bruce Edmonds and Ruth Meyer, 135–171. Berlin: Springer.

- David, Nuno, Jaime Simão, and Helder Coelho. 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). <http://jasss.soc.surrey.ac.uk/8/4/2.html>.
- David, Nuno, Jaime Simão, and Helder Coelho. 2007. "Simulation as Formal and Generative Social Science: The Very Idea." In *Worldviews, Science and Us*, edited by Carlos Gershenson, Diederik Aerts, and Bruce Edmonds, 266–284. Singapore: World Scientific.
- Denning, Peter. 2010. "The Great Principles of Computing." *American Scientist* 98 (5): 369–72.
- Edmonds, Bruce. 2000. "The Use of Models - Making MABS More Informative." In *Multi-Agent-Based Simulation*, edited by Scott Moss and Paul Davidsson, 15–32. Berlin: Springer.
- Epstein, Joshua. 1999. "Agent-Based Computational Models and Generative Social Science." *Complexity* 4 (5): 41–60.
- Evans, Michael. 1984. *Productive Software Test Management*. New York: Wiley.
- Fetzer, James. 2001. *Computers and Cognition: Why Minds Are Not Machines*. Dordrecht: Springer.
- Fraassen, Bas van. 2008. *Scientific Representation*. Oxford: Oxford University Press.
- Frigg, Roman, and James Nguyen. 2017. "Models and Representation." In *Springer Handbook of Model-Based Science*, edited by Lorenzo Magnani and Tommaso Bertolotti, 49–102. Berlin: Springer.
- Galán, José, Luis Izquierdo, Segismundo Izquierdo, José Santos, Ricardo del Olmo, Adolfo López-Paredes, and Bruce Edmonds. 2009. "Errors and Artefacts in Agent-Based Modelling." *Journal of Artificial Societies and Social Simulation* 12 (1). <http://jasss.soc.surrey.ac.uk/12/1/1.html>.
- Galison, Peter. 1987. *How Experiments End*. Chicago: University of Chicago Press.
- Gilbert, Nigel. 2008. *Agent-Based Models*. London: Sage.
- Gilbert, Nigel, and Klaus Troitzsch. 2005. *Simulation for the Social Scientist*. Glasgow: Open University Press.
- Graebner, Claudius. 2018. "How to Relate Models to Reality? An Epistemological Framework for the Validation and Verification of Computational Models." *Journal of Artificial Societies and Social Simulation* 21 (3). <http://jasss.soc.surrey.ac.uk/21/3/8.html>.
- Guala, Francesco. 2002. "Models, Simulations, and Experiments." In *Model-Based Reasoning: Science, Technology, Values*, edited by Lorenzo Magnani and Nancy Nersessian, 59–74. New York: Kluwer.
- Knuutila, Tarja. 2021. "Imagination Extended and Embedded: Artifactual versus Fictional Accounts of Models." *Synthese* 198 (21): 5077–5097.
- Louie, Marcus, and Kathleen Carley. 2008. "Balancing the Criticisms: Validating Multi-Agent Models of Social Systems." *Simulation Modelling Practice and Theory* 16 (2): 242–56.
- Mcnamara, Laura, Timothy Trucano, George Backus, Scott Mitchell, and Alexander Slepoy. 2008. *R&D for Computational Cognitive and Social Models: Foundations for Model Evaluation through Verification and Validation*. Albuquerque: Sandia National Laboratories.
- Morgan, Mary. 2003. "Experiments Without Material Intervention: Model Experiments, Virtual Experiments and Virtually Experiments." In *The Philosophy of Scientific Experimentation*, edited by Hans Radder, 216–235. Pittsburgh: University of Pittsburgh Press.
- Müller, Peter, and Hans von Storch. 2004. *Computer Modelling in Atmospheric and Oceanic Sciences: Building Knowledge*. Berlin: Springer.
- Nardin, Luis, Giulia Andrighetto, Rosaria Conte, Áron Székely, David Anzola, Corinna Elsenbroich, Ulf Lotzmann, Martin Neumann, Valentina Punzo, and Klaus Troitzsch. 2016. "Simulating Protection Rackets: A Case Study of the Sicilian Mafia." *Autonomous Agents and Multi-Agent Systems* 30 (6): 1117–1147.
- Nersessian, Nancy, and Miles MacLeod. 2017. "Models and Simulations." In *Springer Handbook of Model-Based Science*, edited by Lorenzo Magnani and Tommaso Bertolotti. Berlin: Springer.
- Norling, Emma, Bruce Edmonds, and Ruth Meyer. 2013. "Informal Approaches to Developing Simulation Models." In *Simulating Social Complexity*, edited by Bruce Edmonds and Ruth Meyer, 39–56. Berlin: Springer.
- Pool, Robert. 2011. "Modeling Sociocultural Behavior." In *Sociocultural Data to Accomplish Department of Defense Missions*. Washington D.C.: National Academies Press.
- Powell, Stephen, and Thomas Willemain. 2007. "How Novices Formulate Models. Part I: Qualitative Insights and Implications for Teaching." *Journal of the Operational Research Society* 58 (8): 983–95.
- Quine, Willard. 2013. *Word and Object*. Cambridge: MIT Press.
- Rand, William, and Roland Rust. 2011. "Agent-Based Modeling in Marketing: Guidelines for Rigor." *International Journal of Research in Marketing* 28 (3): 181–93.
- Robinson, Stewart. 2011. "Conceptual Modeling for Simulation: Definition and Requirements." In *Conceptual Modeling for Discrete-Event Simulation*, edited by Stewart Robinson, Roger Brooks, Kathy Kotiadis, and Durk-Jouke van der Zee, 3–30. London: CRC Press.
- Robinson, Stewart. 2020. "Conceptual Modelling for Simulation: Progress and Grand Challenges." *Journal of Simulation* 14 (1): 1–20.
- Sargent, Robert. 2013. "Verification and Validation of Simulation Models." *Journal of Simulation* 7 (1): 12–24.

- Scherer, Sabrina, Maria Wimmer, Ulf Lotzmann, Scott Moss, and Daniele Pinotti.** 2015. "Evidence Based and Conceptual Model Driven Approach for Agent-Based Policy Modelling." *Journal of Artificial Societies and Social Simulation* 18 (3). <http://jasss.soc.surrey.ac.uk/18/3/14.html>.
- Schulze, Jule, Birgit Müller, Jürgen Groeneveld, and Volker Grimm.** 2017. "Agent-Based Modelling of Social-Ecological Systems: Achievements, Challenges, and a Way Forward." *Journal of Artificial Societies and Social Simulation* 20 (2). <http://jasss.soc.surrey.ac.uk/20/2/8.html>.
- Shadish, William, Thomas Cook, and Donald Campbell.** 2002. *Experimental and Quasi-Experimental Designs for Generalized Causal Inference*. New York: Houghton Mifflin.
- Shiflet, Angela, and George Shiflet.** 2014. *Introduction to Computational Science*. Princeton: Princeton University Press.
- Shiner, Larry.** 2012. "Blurred Boundaries'? Rethinking the Concept of Craft and Its Relation to Art and Design." *Philosophy Compass* 7 (4): 230–44.
- Squazzoni, Flaminio.** 2012. *Agent-Based Computational Sociology*. London: Wiley.
- Tedre, Matti.** 2015. *The Science of Computing*. London: CRC Press.
- Troitzsch, Klaus.** 2016. "Extortion Rackets: An Event-Oriented Model of Interventions." In *Social Dimensions of Organised Crime*, edited by Corinna Elsenbroich, David Anzola, and Nigel Gilbert, 117–132. Berlin: Springer.
- van der Zee, Durk-Jouke, Andreas Tolk, Mike Pidd, Kathy Kotiadis, and Antuela Tako.** 2011. "Education on Conceptual Modeling for Simulation – Beyond the Craft: A Summary of a Recent Expert Panel Discussion." *SCS Me&S Magazine* 2.
- Wang, Wang, and Roger Brooks.** 2011. "Improving the Understanding of Conceptual Modeling." In *Conceptual Modeling for Discrete-Event Simulation*, edited by Stewart Robinson, Roger Brooks, Kathy Kotiadis, and Durk-Jouke van der Zee, 57–70. London: CRC Press.
- Willemain, Thomas.** 1994. "Insights on Modeling from a Dozen Experts." *Operations Research* 42 (2): 213–22.
- Winsberg, Eric.** 2010. *Science in the Age of Computer Simulation*. Chicago: University of Chicago Press.
- Wolfram, Stephen.** 1984. "Computer Software in Science and Mathematics." *Scientific American* 251 (1): 188–203.

David Anzola is an associate professor in the School of Management at Universidad del Rosario (Colombia) and a member of the Innovation Centre. His research interests are in the philosophy of computational social science, complexity, digital technologies, and social theory and metatheory. He is currently working on the methodology and social epistemology of agent-based social simulation.