Chapter 37
What is a Computer Simulation and What does this Mean for Simulation Validation?

Claus Beisbart

Abstract Many questions about the fundamentals of some area take the form "What is …?" It does not come as a surprise then that, at the dawn of Western philosophy, Socrates asked the questions of what piety, courage, and justice are. Nor is it a wonder that the philosophical preoccupation with computer simulations centered, among other things, about the question of what computer simulations are. Very often, this question has been answered by stating that computer simulation is a species of a well-known method, e.g., experimentation. Other answers claim at least a close relationship between computer simulation and another method. In any case, correct answers to the question of what a computer simulation is should help us to better understand what validation of simulations is. The aim of this chapter is to discuss the most important proposals to understand computer simulation in terms of another method and to trace consequences for validation. Although it has sometimes been claimed that computer simulations are experiments, there are strong reasons to reject this view. A more appropriate proposal is to say that computer simulations often model experiments. This implies that the simulation scientists should to some extent imitate the validation of an experiment. But the validation of computer simulations turns out to be more comprehensive. Computer simulations have also been conceptualized as thought experiments or close cousins of the latter. This seems true, but not very telling since thought experiments are not a standard method and since it is controversial how they contribute to our acquisition of knowledge. I thus consider a specific view on thought experiments to make some progress on understanding simulations and their validation. There is finally a close connection between computer simulation and modeling, and it can be shown that the validation of a computer simulation is the validation of a specific model, which may either be thought to be mathematical or fictional.

Keywords Definition · Experiments · Thought experiments · Argumentation · Models · Internal vs. external validity

---

C. Beisbart (✉)
Institute of Philosophy, University of Bern, Bern, Switzerland
e-mail: Claus.Beisbart@philo.unibe.ch

C. Beisbart and N. J. Saam (eds.), *Computer Simulation Validation*,
Simulation Foundations, Methods and Applications,
https://doi.org/10.1007/978-3-319-70766-2_37

37.1 Introduction

What is validation of computer simulations and how does it work? One strategy to make progress on these questions is to put another, apparently more fundamental, question first: What is a computer simulation, to begin with? The idea is that a closer understanding of what computer simulation is usefully constrains any sensible view on its validation.

The aim of this chapter is to pursue this strategy. I will thus address the question of what a computer simulation is and then consider consequences for understanding validation. The question of what computer simulation is has indeed been at the center of a lively philosophical debate (see Imbert 2017 and Saam 2017 for overviews). I will draw on this debate and consider important proposals about what a computer simulation is.

The question of what computer simulation is can most naturally be answered in terms of a definition (see Gupta 2015 for a primer on definition). In the recent philosophical literature about computer simulations, we do find attempts at such a definition (Hartmann 1996, Sect. 2 and Humphreys 2004, pp. 110–114). But the question of what computer simulation is has often been answered in a loser sense by subsuming it under, or relating it to, some known method such as experiment. In this chapter, I concentrate exclusively on proposals that spell out what computer simulations are by claiming a close association between computer simulation and some other method. The reason is that such accounts seem particularly promising for a better understanding of the validation of simulations because they open pathways into known territory. I do not require that the proposals under consideration aim at a full-fledged definition of simulation. Some proposals that have been much discussed in the literature do not attempt to give such a definition, and it seems inappropriate to exclude them. There is of course a downside, when I include accounts that do not provide a full definition and that do not even typify simulations in terms of a genus: The accounts are weaker and less informative. But more precision as to what a simulation really is would not likely be of much help for the understanding of validation. It has in general proven difficult to specify what sort of things other artifacts or creations of the human mind, e.g., novels or symphonies, are precisely. Fortunately, the type of thing to which artifacts belong seems rather immaterial for other questions we may ask about them.[1]

In this chapter, I will thus go through a number of *methods*, ask whether computer simulation may be understood by relating it to the method and then trace consequences for validation. Now a specific proposal to the effect that computer simulation is closely associated with such and such method will not deepen our understanding of the validation of simulations, if the proposal itself is implausible.

---

[1] If somebody claims that computer simulations are, say, experiments, then what is claimed may either be regarded as essential of computer simulation (such that it should be included in its definition), or it may be supposed to be a contingent claim about computer simulations. I take it that the views under consideration are meant to capture essential properties of simulations, but this is not necessary for my argument.

So I'll briefly evaluate each proposal under consideration. My conclusions in this respect may not be shared by every author in the field because the nature of computer simulations has remained controversial. I nevertheless hope to be fair and give most positions due consideration. For reasons of space, the discussion has to be brief, so I cannot fully cover the existent literature. Note further that the consequences that the accounts under consideration have for validation haven't yet been worked out; so in this respect, this chapter will move beyond the existent literature.

After some preliminaries, I'll start with the method of experiment (Sect. 37.3), move to thought experiment (Sect. 37.4) and modeling (Sect. 37.5), before I conclude in Sect. 37.6. Views that take computer simulations to be genuinely novel (see, e.g., Winsberg 2001; see Frigg and Reiss 2009 for discussion) are not covered in this way, but they do not promise an easy route to better understanding validation, so we can bracket them in what follows (in this volume, Chap. 43 by Imbert addresses the questions of whether, and how, computer simulation is novel).

37.2 Preliminaries

Before I look at various proposals that relate computer simulation to other methods, it is useful to comment on two important concepts, viz., those of computer simulation and of validation. As already implicit in the discussion so far, I assume that computer simulation qualifies as a scientific method. A scientific method, in turn, is a type of activity that scientists engage in to promote the ultimate aims of science, e.g., to gain knowledge and understanding. For the purposes of this chapter, I take it that a computer simulation crucially involves the run of a simulation program that provides possibly partial and approximate solutions to equations that trace the dynamical evolution of a target system (this is close to Humphreys' definition 2004, pp. 110–114). In this way, the dynamical evolution of the target is imitated or modeled. There are different ways to specify in more detail what sort of activities form part of a computer simulation (see, e.g., Parker 2009, p. 488), but how exactly this is done will not matter for our purposes. Simulations that do not involve a digital computer, e.g., analogue simulations, are neglected for the purposes of this chapter.

Regarding the notion of validation, we have to be very careful. In the sciences, not only computer simulations are said to be validated; rather, scientists talk about the validation of models and experiments too. This suggests that there is a general idea of validation that covers more specific notions like the validation of experiment, etc. In what follows, validation in this general sense will be called validation$^{\mathrm{gen}}$. It comprises, very roughly, the activities that make a case that the results that have been, or can be, obtained by applying the method in a particular case do in fact hold true of one or more real-world systems. It is important to note that the activities of validation$^{\mathrm{gen}}$ refer to reality.

In my discussion, I will sometimes focus on the validation of few actual results that have been obtained by applying the method in a specific case. But my considerations are meant to carry over to a more comprehensive validation. The latter covers not only the results that have in fact been obtained but also results that could be obtained by applying the method in a specific case, e.g., by running the same experiment in slightly different ways. I allow that validation^{gen} can also be concerned with the question of whether the assumptions built into a specific application of the method hold.

As far as the validation of computer simulations, validation^{cs}, is concerned, I assume that it is validation^{gen}, as applied to computer simulations. In this case, the results are constructed from the output, which consists of values of characteristics such as position, energy, etc. The results can be cast in claims about the target system, which are either quantitative or qualitative. In the simplest case, such a claim has it that certain characteristics of the target system, say, the luminosity of a star, has such and such value, as output from the simulation. Since the outputs from simulations are affected by all kinds of errors, the output numbers reflect the target system at best only up to some accuracy. The question of whether the results of a simulation hold thus is meant to be the question of whether the results are sufficiently accurate for the intended applications. The point of validation then is to show that the results are sufficiently accurate. This understanding of simulation validation accords well with the famous definition of validation^{cs} by Schlesinger et al. (1979, p. 104).

My focus in this chapter is on validation^{cs} as applied to a simulation program or the model implicit in it, call it the computational model. At this point, it does not make a difference whether we talk about the program or the computational model because the program delivers exact solutions to the computational model (this is just how the latter is defined). The computational model needs to be distinguished, however, from the conceptual model which is typically the scientific model that scientists are interested in before they run the simulations. In many simulations, this model consists of differential equations, while the computational model approximates the latter in some way. So-called verification is supposed to show that the computational model is a faithful representation of the conceptual model for the purposes of the inquiry. If a simulation is properly verified, then the distinction between the conceptual and the computational model does not much matter. If it is not clear whether the simulation is verified, then we can in principle distinguish the validation of the conceptual model, validation^{con}, from that of the computational model (or the computer program), which has been called validation^{cs}.

Now if computer simulation turns out to be closely related to another method, then it is likely that we obtain consequences for validation^{cs}. In particular, if computer simulation is intimately connected to a method for which validation is an issue, it may turn out that validation^{cs} boils down to the validation of the method to which computer simulation is assimilated. Since we will examine several distinct attempts to relate computer simulations to other methods, it is likely that we will consider different, possibly inconsistent claims that arise for validation^{cs}. It may be thought that this leads to different concepts of validation^{cs}, for instance, the concept of validation^{cs,e} (validation of computer simulation under the assumption that the latter is an experiment), etc. In what follows, I will refrain from distinguishing such concepts because we are ultimately interested in one concept, viz., that of validation^{cs}, and since it is possible to discuss various views on validation$^{\mathrm{cs}}$ without assuming several concepts. Further, as I will briefly argue in Sect. 37.6 below, the proposals that prove to be sensible in our discussion are compatible with each other.

37.3 Computer Simulations and Experiments

Computer simulations are often called computer experiments (e.g., Beeler 1983), where the term "experiment" is sometimes put in scare quotes (e.g., in the title of Verlet 1967). Likewise, the term "experiment in silico" is used for simulations (e.g., Naumova et al. 2008). All this is no accident. There are in fact close parallels between experiments and computer simulations (see Beisbart 2018, Sect. 3). As is well known, experiments crucially involve two types of causal interaction between the working scientist and the system she is working on: intervention—an experimental system is set up or at least manipulated in some way—and observation—the reaction of the system is observed (see, e.g., Heidelberger 2005; Radder 2009 and Franklin and Perovic 2016 for reviews about experiment). Computer simulations seem to function in a parallel way: Simulation scientists interfere with the hardware of a computer to set up a system, which is then investigated. After the program has been run, they obtain outputs that are interpreted in the same way as are data from observation of an experiment. It is no surprise then that some philosophers have tried to understand computer simulation in terms of experiment.

37.3.1 Computer Simulations as Experiments

Some authors have gone as far as to claim that computer simulations are, or crucially involve, experiments. In more detail, there are two ways in which this claim may be spelled out (Beisbart 2018, Sect. 4): Either *the computer* itself is supposed to be the system experimented on. The results obtained for this system are then transferred to the target system of the simulation, e.g., to a cell or a galaxy that is simulated. Parker (2009) defends this view. She takes it to be obvious that a simulation involves the twofold causal interaction with a computer that is constitutive of experiment (ibid., p. 488). She further argues that the experimental status of simulations is important for a comprehensive epistemology of computer simulation because certain concerns that may arise about a simulation hinge on the fact that a material system (here, the computer) is under investigation (pp. 489–491). As an alternative view, it is suggested that at least some simulations are really experiments *on the target system* of the simulation (e.g., a galaxy). Morrison (2009) takes some steps in this direction, although she never claims simulations to be experiments. But she definitely takes some simulations and experiments to be on par epistemically, e.g., because models function in the same way in both methods. Massimi and Bhimji (2015) make a stronger case for the view that simulations are experiments by arguing that some computer simulations from particle physics involve causal interactions that are not relevantly different from the causal interactions between experimenters and the systems they experiment on (cf. also Morrison 2015, Part III).

Suppose now, for the sake of argument, that at least some computer simulations are, or crucially include, experiments. What would this mean for validation^{cs}?

It is first interesting to note that the term "validation" or "validity" is well-established for experiments too. In some sciences, e.g., the social sciences, it is common to distinguish between internal and external validity^{e}.[2] For our purposes, we may understand the distinction as follows: Internal validity^{e} is about results that concern the system with which the experimenter interacts causally (i.e., the system experimented on) during the time when the experiment is run. External validity^{e}, by contrast, is about the generalization to other times, condititions, systems, etc. For a simple example, an experimentalist may suspect that, in the system on which she has experimented, a particular medical treatment of a person has caused her recovery. The experiment is internally validated^{e} in this respect, if the experimenter shows that the treatment did cause the recovery in the specific case under consideration. The experiment is externally validated^{e} if the effect can be shown to generalize to other patients. Both ways, validity^{e} is a matter of inference. Note that external validity^{e} is only a concern if scientists wish to generalize their results. This condition is not met in all experiments. It is possible to run an experiment on, say, a population of animals just to learn about this very population at one time.

The distinction between internal and external validity^{e} is quite rough. In some areas, other, more fine-grained distinctions are drawn. For instance, in educational and psychological research, people discriminate between construct, content and criterion validity^{e} (see, e.g., Newton and Shaw 2014). Here, construct validity^{e} is roughly supposed to ensure that a measurement does indeed reflect a theoretical construct. In what follows, we cannot discuss such domain-specific notions of validity^{e}, but we will below comment on construct validity^{e} and relate it to validity^{cs}.

If simulations were, or included, experiments *on the computer hardware*, they would be internally valid^{e} qua experiment if results about the computer were shown to be genuine (this is in fact suggested by Parker 2008, p. 168). But when running a computer simulation, scientists do not typically establish any results about the computer hardware. The outputs of the computer simulation are immediately interpreted in terms of the target system, and not in terms of the computer hardware. For instance, if a simulation program prints a series of numbers, the latter are interpreted as temperatures of the target system for a series of times. In fact, most simulation scientists cannot even use the output to infer anything interesting about the hardware because they know virtually nothing about the hardware. So assuming that computer simulations are experiments, internal validity^{e} is not a matter of concern for computer simulations. But then, nor can external validity^{e} be. The point of external validity^{e} is the generalization of results that have been established about the system

---

[2] The distinction goes back to Campbell 1957 (see Winsberg 2009, p. 579 following Parker). See also Campbell and Stanley (1963, p. 5) and Cook and Campbell (1979, p. 37). It was originally restricted to experiments in social science that aim at causal claims.

experimented on. If no such results have been obtained, external validity^e cannot get started.

So the notions of internal and external validity^e do not make much sense for computer simulations, if the latter are considered to be experiments on the hardware. This casts doubts on the very idea that simulations are experiments on the hardware. It is in fact problematic to say that computers are observed qua experimental system if working simulation scientists only understand the output in terms of the target system. Likewise, it seems problematic to suggest that computers are manipulated in the way experimental systems are, if the working scientists don't really know what they are doing with the computer qua material system when they, e.g., type commands in the keyboard. For these reasons, computer simulations are not, and do not include, experiments on the hardware of the computer (see Beisbart 2018, Sect. 5 for details).

Turn now to the proposal that computer simulations are experiments *on the target*. This view fits better with the distinction between internal and external validation^e of experiments. Suppose for instance that a merger between two known galaxies is simulated. Qua experiment, the simulation would be internally valid^e, if it produced genuine results about the specific galaxies involved in the merger. It would be externally valid^e, if the results were shown to extend to other mergers of galaxies.

The problem though is that the proposed view, viz., that computer simulations are experiments on the target, itself isn't plausible. First and quite obviously (and pace Massimi and Bhimji 2015), computer simulations do not involve the characteristic twofold causal interaction between the experimenter and the experimental system. If a galaxy is simulated, this system is neither manipulated nor observed (see, e.g., Beisbart 2018, Sect. 6). Second, there is a crucial epistemological difference between computer simulations and experiments in that the assumptions built into a simulation in some sense imply the results, whereas this is not so for experiments (Arnold 2013, pp. 59–60, Beisbart 2018, Sect. 6). A consequence is that, in the words of Morgan (2005, p. 324), whereas simulations may surprise, only experiments can confound and thus lead scientists to question their assumptions.

37.3.2 Computer Simulations as Modeled Experiments

Now if computer simulations are neither experiments on the hardware nor on the target of a simulation, how can we explain the striking similarities between both methods? One proposal is that computer simulations can *model* possible experiments and do in fact often do so (cf. the title of Winsberg 2003; see Beisbart 2018 for an elaboration). The idea is that simulations first allow the scientist to model interventions on the target by setting the initial conditions and the parameter values that serve as input to a simulation program. The reaction of the target then is traced by running the program. Finally, observation and analysis of the data are modeled by those activities with which simulation scientists process the output of the simulations. In this way, computer simulations allow the representation of possible experiments.

I'm here talking of *possible* experiments because the experiments that are represented may in fact never be carried out on the real target system. Note further that some actual computer simulations (qua runs of a simulation program) do not model an experiment because they just try to represent the dynamics of a real-world target system as it happens to be like without any intervention. It is finally important to note that the various steps of an experiment are modeled in different ways: While the reaction of the target system is just modeled using the model implicit in the computer simulation, the intervention on the target system and the observation are not (the program does not contain variables that trace the working scientist looking at the target system, for instance). Rather, intervention and observation are modeled by activities on the part of the computer scientist, when she sets the initial conditions and observes the result on the screen.

If this proposal is on the right track (and it may be debated whether it is), then simulation scientists should appropriately model activities that establish internal and, if applicable, external validation^{e}. This consequence is not implausible. To show this, we consider a simple schematic example. We concentrate on internal validity^{e}, because it is generic. So suppose that nano-scientists want to know how the flow of a fluid through a nano-channel is influenced by the roughness of the wall of the channel. They run a molecular dynamics simulation of the system (see Liu et al. 2010 for an example of such a simulation). Suppose that their simulation outputs indicate that "roughness reduces the electro-osmotic flow rate dramatically even though the roughness is very small compared to the channel width" in a specific case (ibid., p. 7834). In an analogous way, measurements from an experiment may in principle indicate such an effect. In what follows we assume that this (qualitative) claim is the result that the scientists are interested in. Now to internally validate^{e} this result, experimentalists need to show that the effect is genuine. For instance, they have to make sure that there is in fact a dramatic reduction of the flow rate. Also, since the reported result has some generality because there are many ways in which the surface of a wall may be rough, the experiment has to be run for different realizations of roughness. But if this is needed in the experiment, then simulation scientists should model this internal validation^{e}, when they run the simulation. That is, they need to make sure that the simulation output does in fact imply a dramatic reduction of the flow rate and that the effect holds for many realizations of roughness. They will thus run the program with different parameters for the roughness. These activities form certainly some part of validation^{cs} because they are needed to show that there is this effect in the target. So we can say that, to some extent, validation^{cs} models internal validation^{e}.

Note though that certain concerns that matter for the validation^{e} of experiments are often not a real issue in the validation^{cs} computer simulations: For instance, in many experiments, some characteristics are measured using extremely complicated measurement devices. Internal validation^{e} has to make sure that the measurement devices function as intended. And it's a matter of construct validity^{e} that the measurements do in fact reflect the construct scientists are interested in. In many simulations, all this is not an issue because the characteristics are traced by the computer simulation program such that their values can be output and directly inspected by the scientists (there are some simulations that cover the measurement devices too, for instance in particle physics; see Massimi and Bhimji 2015 for a philosophical account). For another issue that need not concern simulation scientists, experimentalists cannot perfectly shield their experiments from external influences. If they observe a specific effect, they have to exclude that it was produced by external factors not controlled for in the experiment. This is not a concern for most simulations because they typically isolate the system under consideration in a perfect manner simply by not modeling external factors.[3]

Conversely, there is also a task in validation^{cs} that does not have a counterpart in validation^{e}: Simulation scientists need to show that their simulation faithfully traces the behavior of the real-world target system. If it doesn't, then what is claimed as result doesn't hold. The focus here is on the reaction of the target system. The computer program may after all misrepresent the way in which the target system behaves under the conditions that have been set. The proposal that computer simulations can model experiments can account for such practices. The reason is that, under the proposal, the experiment is only modeled and the model needs of course validation^{m} too (see below for validation^{m} of models).

Thus, assuming the proposal, we can distinguish between two layers of validation^{cs}: First, it must be shown that the modeled experiment really has such and such as result. This is to model internal validation^{e} from the experiment. As indicated, this is typically much easier than for real experiments. Second, it must be shown that the model of the experiment delivers a faithful representation of the way in which the target system reacts to the setup produced initially. This covers most part of the validation^{cs}, as it is known for simulations.

Now when simulation scientists validate^{e} their simulation, this closely resembles the validation^{e} of experiments. As Parker (2008) argues, at least five validation^{e} strategies known from the validation^{e} of experiments have close parallels in the validation^{cs} of computer simulation. For instance, both experimentalists and simulationalists can argue that their apparatus/simulation is built upon well-confirmed theory. Most importantly, both experimentalists and simulationalists can choose the so-called Sherlock Holmes strategy, which is to exclude all sorts of errors. Parker is certainly right in observing such parallels. Maybe, they can to some extent be accounted for by saying that the simulations model experiments. But the parallels should not lead us to assimilate validation$^{e/cs}$ of experiments and of computer simulations too much. As noted above, only the latter has to make a case that the reaction of the target system is faithfully modeled. This point is also clear from Parker's discussion when she notes that a simulation may be validated^{cs} by validating^{m} the underlying model and then showing that the computer program does in fact yield approximate solutions to the model (ibid., pp. 166–7). Showing that a program delivers such suitable approximations doesn't have a parallel in experiments. When even this part of validation^{cs} works in a similar way as does the validation^{e} of experiments, the reason is not a deeper parallel between computer simulation and experiment, but

---

[3] In a computer simulation, the computer hardware may of course be subject to influences not controlled for. But this is typically excluded by activities of verification, see below.

rather that prescriptions such as “check that there are no errors” apply quite generically to situations in which many errors are possible (ibid., pp. 178–179).

If validation does in fact differ between experiment and computer simulation, we may use validation to discriminate between both methods. This strategy has been adopted by Winsberg (2009). He assumes that experiment and simulation involve an inference from the system that is directly studied to a target system that is typically different from the system studied first. His proposal is (ibid., p. 586):

> what distinguishes simulations from experiments is the *character of the argument given* for the legitimacy of the inference from object to target and the *character of the background knowledge* that grounds that argument.

Very roughly, the crucial idea is that the arguments used in the validation$^{\text{cs}}$ of computer simulation draw on trust that the working scientist has the right sort of principles for modeling the target system under consideration (ibid., p. 587). This is compatible with the view that computer simulations can model experiments.

To sum up then this section: Experimentation and computer simulation resemble each other in many respects. This cannot be explained by saying that the latter are, or include, experiments since simulations do not obey the conditions constitutive of experiments. Rather, many simulations can be said to model possible experiments. The consequence for validation$^{\text{cs}}$ is that, to some extent, the validation$^{\text{cs}}$ of computer simulations may be understood as modeling the validation$^{\text{e}}$ of a possible experiment. But this does not exhaust the validation$^{\text{cs}}$ needed for computer simulations. Rather, the validation$^{\text{cs}}$ of computer simulation needs also to show that the possible experiment is after all well traced. As a matter of coincidence, even this part of validation$^{\text{cs}}$ follows general strategies that are used in experimentation such as the exclusion of errors.

37.4 Computer Simulations, Thought Experiments and Argumentation

If computer simulations are not really experiments, they may still qualify as *thought experiments*. Very roughly, when a scientist runs a thought experiment, she considers a certain scenario and tries to anticipate in thought what will happen in this scenario. For instance, Einstein used a thought experiment involving a train running through a station to show that different observers do not agree on whether two events are simultaneous or not (the thought experiment is described in Einstein 1920, pp. 11–27; see Brown and Fehige 2017 for more examples and a review of the philosophy of thought experiments).[4]

Thought experiments do not involve any causal interaction of the scientist with the system investigated. They are thus not a subclass of experiments. Accordingly,

---

[4] Our focus in this section is exclusively on *scientific* thought experiments. Philosophers too engage in thought experimentation, but it is at least arguable that thought experimentation in philosophy and the sciences function quite differently.

crucial objections against the view that computer simulations are experiments do not apply anymore. It is indeed plausible to say that, in a computer simulation, a certain scenario is thought through as it is in a thought experiment. The role of the computer here is to expand the human capacities to think (see Humphreys 2004, Chap. 1 for this idea).

It does not come as a surprise then that the philosophical literature has closely associated computer simulations and thought experiments. Humphreys (2004, p. 115) has noted that computer simulations have taken the role that thought experiments had in less technologically advanced times. El Skaf and Imbert (2013) argue that thought experiments and simulations alike fall under the same general description of unfolding a scenario.[5] Beisbart (2012) argues that computer simulation and thought experiments fall under the broader category of scientific argumentation, although they differ in a couple of respects. Lenhard (2011), by contrast, draws a starker contrast between the methods and argues that thought experiments have a transparency that computer simulations lack.

For the purposes of this chapter, we need not take a stance on whether computer simulations qualify as thought arguments or whether they are species of the same genus. For even if a close connection between both methods can be established, this connection does not much help understanding computer simulation and its validation^{cs}, unless more is said about thought experiments, and this is in fact difficult. If thought experimentation is a method of its own at all, it is quite peculiar. It is not a method that is applied as widely or as standardly as is experimentation. The examples of scientific thought experiments identified in the philosophical literature are few. And there is no established methodology of running and validating thought experiments.

From a philosophical perspective, it is controversial how thought experiments achieve their tasks. The last two decades have seen a lively philosophical debate on this topic with a wide spectrum of positions. To mention the most extreme ones, whereas Norton (1996, 2004a, b) claims that thought experiments are arguments, Brown (1991, 2004) thinks that some type of thought experiment provides a priori epistemic access to laws of nature. Other positions hold that at least some thought experiments rely on quasi-observational intuitions that can provide justification for belief (Gendler 2004) or that thought experimenting is based upon mental modeling (Nersessian 1992, 2007; see Sect. 5 below for more on models).

So thought experimentation is neither a particularly well-established method nor well understood, as far as its philosophical account is concerned. It will thus not much enhance our understanding of computer simulation if we establish a close connection between the latter and thought experimentation. We will thus turn to a particular philosophical account of thought experiments that promises at least some insight into the validation^{cs} of computer simulations, viz., Norton's so-called argument view. As already indicated, the view has it that thought experiments form a species of arguments and thus instantiate scientific inference. Norton makes a case for

---

[5] They also include experiments under this description, but in the last section, we have already noted crucial differences between experiment and simulation.

this view by, e.g., reconstructing known thought experiments in terms of arguments (e.g., Norton 1996, Sect. III).[6]

For our purposes, it is interesting to note that, in defending his view, Norton addresses potential problems with thought experiments. For one thing, he draws the attention to pairs of thought experiments that yield incompatible results. So at least one of the thought experiments must be deficient. If thought experiments are arguments, the incompatibility of their results can be explained by saying that at least one of the underlying arguments is not sound (Norton 2004b, Sect. 3). For another thing, Norton claims that the past record of thought experimentation is not impressive because many thought experiments have arrived at a wrong conclusion. He thus demands a mark of reliability for thought experiments. His own proposed mark is that the form of a thought experiment is taken to be legitimate by some logic (Norton 2004b, Sect. 4).[7] Finally, he suggests that a reconstruction of a thought experiment in terms of an explicit argument may clarify its merits, e.g., by uncovering hidden premises (Norton 1996, Sect. 3.1).

So the argument view has some resources to address the reliability of thought experiments and, maybe, to develop a related methodology. Argumentation is in fact some part of the scientific method and has been extensively studied in logic and philosophy of science. The findings and techniques from logic in particular are of great help in the assessment of arguments. If the argument view of thought experiments has it right, logic may turn out helpful in the assessment of thought experiments too.

Let us thus try to extend the argument view to computer simulations (see Beisbart 2012 for details). The rough idea is that each run of a computer simulation program goes through some argument. The premises of the argument are the assumptions that underlie the simulation, e.g., about the dynamics of the target system or about the initial conditions. The conclusions are the results, which can be obtained from the output of the simulation. They have it that certain characteristics (temperature, positions of particles, etc.) take these and these values. Clearly, if all goes well, the conclusions follow from the model assumptions. So each simulation can at least be reconstructed as an argument. If we adopt the extended mind hypothesis (Clark and Chalmers 1998), we can even show that a coupled system consisting of the working scientist and the computer runs through the argument as a matter of fact.

Set up in this way, the argument view about computer simulations has some plausibility. When we were discussing above whether computer simulations are experiments, we have argued that they are not because the model assumptions implicit in the simulation imply in some way the result. This is to say that there is an argument running from the model assumptions to the result of a simulation. The argument view is focused on this very argument.

---

[6] There is no need here to draw on Nersessian's view that thought experimenting involves mental modeling since we'll examine simulations and models in due course in Sect. 37.4.

[7] This mark is not sufficient for a good thought experiment because even a valid argument can arrive at a wrong conclusion if some premise is false. But this complication does not matter for our argument.

Suppose now that this view is on the right track. What would the implications for the validation^{cs} of computer simulations be?[8] Well, if a computer simulation is an argument with the result as a conclusion, the result is likely true if i. the premises are likely true and ii. if they strongly support the conclusion. This suggests a certain two-step strategy to validation^{cs}.

How exactly this strategy to the validation^{cs} of computer simulation looks like turns on what exactly we take the argument and its premises to be. From the viewpoint of working scientists, it is natural to say that the argument takes the assumptions from the conceptual model as premises. After all, it is the conceptual model that is at the center of what scientists think about the target system. Thus, if we assume that the argument starts from the assumptions of the conceptual model as premises, then, in the first step in the strategy to validate^{cs} the simulation, scientists need to check that the assumptions of the conceptual model are likely true. For instance, the premises may be considered likely true either because they draw on well-confirmed theory or because they report measurements (e.g., about the initial conditions or of some parameter values). Now the assumptions of a conceptual model are often known to be false, because they are based upon approximations and idealizations. But even then, the assumptions may still be sufficiently accurate for the purposes of the inquiry. So if we weaken the premises and let them claim that the model assumptions are to some extent accurate, then scientists may be able to make a case for their likely truth.

In a second step, scientists have to check whether the premises strongly support the conclusion, i.e., what they obtain as result. Now what the computer does in order to produce the result is to go through a number of calculations. Exact calculations can be cast as deductive arguments (in fact, the point of a calculus is to obtain deductive arguments). This would mean that the arguments are as strong as they can be because the truth of the premises guarantees the truth of the conclusions. But in a computer simulation, the computer does not carry out exact calculations about the conceptual model. As is well known, the calculations done by the computer involve all kinds of errors with respect to the conceptual model, e.g., errors that arise from the discretization of differential equations or roundoff errors (see Chap. 5 by Roy in this volume). There may even be hardware failures or programming errors that prevent the computer program from working as intended. Now at least the known errors are usually taken into account when the results are formulated. As indicated in the preliminaries, scientists do not assume that the value of a characteristics in their target system is precisely, say, 4325 in some units, if this is the number output by the computer program. They rather assume that the output number reflects the true value up to some accuracy. Taken in this way, there is a chance that the premises (i.e., the model assumptions) do in fact imply the conclusion (i.e., the result), as is expected for a deductive argument.

But clearly, work is needed to show that the conclusion does in fact follow from the premises in a specific case. For instance, unknown errors due to hardware failure have to be excluded. Further, the accuracy of the results needs to be determined in

---

[8] Baumberger et al. (2017) have recently proposed to frame validation^{cs} using notions from argumentation theory. But this conceptualization of validation^{cs} is independent of the argument view.

such a way that they follow from the fact that the model assumptions hold to some accuracy. And if there are uncertainties in some model assumptions, scientists have to check what uncertainties they produce for the results. All these activities are well known as verification of a computer simulation program. To achieve this, scientists cannot write down an argument and check it using the standards of some logic (as might somehow be suggested by the argument view). This is too complicated because too many calculations are involved in a simulation. The reason is that computer simulations are opaque: We cannot see how the results follow from the premises in the way in which we can do this in simple thought experiments (see Humphreys 2004, Sect. 5.3 and Humphreys 2009, pp. 618–9 for opaqueness).

So far then, the argument view suggests a two-step procedure for validation$^{\mathrm{cs}}$. If a computer simulation is an argument with premises from the conceptual model, then, in a first step, the accuracy of the premises is to be secured. A second step is supposed to make a case that the argument is sufficiently strong such that the conclusions follow from the premises. In principle, this is certainly a sensible approach to validation$^{\mathrm{cs}}$. If validation$^{\mathrm{cs}}$ of a computer simulation is supposed to make a case that the results hold (up to some accuary), then a viable route is to show that the conceptual model that underlies the simulation is sufficiently accurate and that the computer program delivers results that are sufficiently accurate with respect to the conceptual model (see also Chap. 42 by Beisbart in this volume).

But there is a problem with this two-step strategy. The challenge is to make a sufficiently strong case for the premises, i.e., for the assumptions of the conceptual model. Even if the dynamcis of the target system is well understood in terms of a well-confirmed theory, this does not guarantee sufficient accuracy of the conceptual model because the theory needs to be combined with additional assumptions, e.g., about parameter values, initial and boundary conditions to produce a concrete model, and many of these assumptions will at best be uncertain. The only route to make a case for the conceptual model as a whole then is to run the simulation and to see whether the results match with measurements from the target system to a sufficient degree of accuracy. This strategy is of course familiar from validation$^{\mathrm{cs}}$, as it is known and described in the literature. A comparison between data from the target (or, maybe, a system sufficiently similar to the target) and simulation output is what many people take to be crucial about validation$^{\mathrm{cs}}$. This data-oriented approach to validation is not incompatible with the argument view. In terms of this view, what is crucial in the comparison between simulation output and measured data is, very roughly, this: Some premises of the argument behind the simulation are uncertain (e.g., because there is uncertainty about the values of certain parameters), so some of their consequences are derived using a simulation. If these consequences turn out to be true, then the premises are to some extent confirmed, and this confirmation extends to other consequences of the premises in new applications of the program. This way of reasoning is often called hypothetico-deductive approach. It makes use of deductive arguments, but in a way that is not as straightforward as to reason from the premises to the conclusion. A closer analysis of the inferences involved is beyond the scope of this chapter (but see Chap. 42 by Beisbart in this volume).

All in all, the argument view about computer simulations accommodates the activities of validation$^{\mathrm{cs}}$ as follows: It conceptualizes validation$^{\mathrm{cs}}$ in terms of an argument that is used as a reconstruction of the simulation. The basic point of validation$^{\mathrm{cs}}$ then is to argue that this very argument is sound. Because validation$^{\mathrm{cs}}$ is thus an argument about an argument, it can be called a meta-inference. What is a natural suggestion from the viewpoint of the argument view is a separation between the examination of the premises and of the way they support the conclusion (i.e., the results). If the premises are supposed to be assumptions from the conceptual model, then the argument view invites a 2-step procedure: A case is made that the conceptual model is sufficiently accurate (we can call this validation$^{\mathrm{con}}$ of the conceptual model) and a case is made that the results do follow from the premises (this is verification of the simulation program). However, this two-step procedure is often not viable because the conceptual model cannot be validated$^{\mathrm{con}}$ independently from the simulations, and to this extent, the suggestion on behalf of the argument view is not useful. Note though that validation$^{\mathrm{cs}}$ activities that compare simulation results with measured data can be accommodated within the argument view too. Nevertheless, the argument view remains a bit artificial in that the argument that has been proposed as reconstruction does not lend itself to an investigation because it is unclear how the conclusion follows from the premise. The argument is also quite far from the calculations done in the computer, which use all kinds of approximation schemes.[9]

37.5 Models and Simulations

The proposal that there is a close connection between computer simulation and modeling, indeed that computer simulations are some sort of models is now more than just in the air. In Sect. 37.3 above, we have proposed to say that computer simulations can model possible experiments. In Sect. 37.4 we have observed a continuity between simulation and thought experiment, where some authors take the latter to crucially involve mental modeling.

Talk about simulations too indicates a strong link between simulations and models, e.g., when people speak of simulation models. Hartmann's 1996 definition of computer simulation, according to which a simulation forms a process that imitates another process, also establishes a connection to models, insofar as imitation is a sort of modeling or representation. What is further promising for our purposes is that validation is an issue for models too (see, e.g., Koblick 1959, p. 642 for an example).

---

[9] We might also have provided a slightly different argument to represent a run of a computer simulation program: The idea would be that the premises state the computational model. Now the latter is defined such that results of the simulations are exact solutions to the computational model. So it is not an issue anymore to check that the argument is deductive. But the work of validation is only shifted to the examination of the premises. For instance, if we want to make a case that the computational model is sufficiently accurate by drawing on prior commitments to theory, we must show that the theory is likely sufficiently accurate and that it is appropriately reflected in the computational model.

Let us thus try to conceptualize simulations in terms of models and probe possible conclusions for the understanding of validation^{cs}.

A challenge that we face when thinking about models is an embarrassment of riches. There are not just many models, they also belong to different categories (see Frigg and Hartmann 2017 for a philosophical overview). Some models are material systems, e.g., scale models of cities or cars. Other models are merely imagined, e.g., a number of point particles connected by massless springs. Sometimes, a set of equations is said to form a model too. So we have at least material, fictional and mathematical models. There are likely more types of models, but the three categories will suffice for our purposes. Note too that models of different types are often closely related to each other; for example, mathematical equations can describe a fictional system.

In view of the plurality of (types of) models, it will hardly be illuminating to call computer simulations models, unless more is said about models. What then does modeling amount to and is there anything common to all types of models? A first observation that is relevant in this respect is that models are typically based upon simplifications. Various features of the target are abstracted away, idealizations are assumed and approximations made. For instance, a scale model of a city leaves out small-scale decorations of houses, it gives all buildings the same color and approximates the marketplace as a square. This gives rise to the following proposal: A model is a system that is distinct from the target system but used as a surrogate for the latter. Since the model is simpler than the target, scientists can more easily learn about the model; nevertheless, some of the findings obtained for the model can be transferred to the target system. I take this to be the core of insightful philosophical accounts of modeling, e.g., by Hughes (1997), Suárez (2004) and Weisberg (2007). This core suggests that modeling is an indirect research method that takes a detour via a surrogate (Weisberg 2007, p. 207). It further implies that modeling may be split into three stages, viz., construction of the model, analysis of the model and coordination between model and target (ibid., pp. 222–226). This view nicely accounts for material models and fictional models. It is less clear, however, how it applies to mathematical models. We may either stretch words a bit and say that sets of mathematical equations too constitute systems that are studied as surrogates for their targets (cf. Suárez 2004). Alternatively, we may say that mathematical equations are not really models, but model descriptions (see Weisberg 2007, p. 217). We can then say that they specify the dynamics of other, e.g., fictional models. This makes a lot of sense as long as the equations involve significant simplifications. If, by contrast, some equations are not built upon simplifications and iterally hold true of the target system, then we can take them to be descriptions of the target, and we need not call them a model.

Validation^{m} is an issue for models because modelers first obtain results for their models which they then need to translate to their targets. This is not to deny that the issue may arise whether results obtained for a model are genuine. Echoing the distinction known for experiments, we can say that this is a matter of internal model validity^{m}. But issues of internal validity^{m} are not specific to models. For instance, if an experiment on a material model is run, internal validity^{m} of the results about the model is internal validity^{e} of the experimental results. The crucial and characteristic question of validation^{m} is rather external: What is the justification to assume that some results obtained for the model apply to the target too? Note that external validity^{e} of experiments is about a similar sort of inference.

Let us now go back to our main topic and to simulations. What precisely is their relationship to models? This is an intricate issue because computer simulations involve various models of several types (see Beisbart 2014 for a more extensive discussion).

First, each computer simulation crucially involves a mathematical model. This holds true not only of computer simulations that attempt to solve ordinary or partial differential equations. It is also true of, e.g., agent-based models. Such models need not involve variables that take numbers as values, but they nevertheless involve equations, which trace the time evolution of purely qualitative variables (e.g., the preferred political party). More generally, every simulation contains rules that are supposed to trace the dynamics of the target system in some respect, and these rules can be cast as mathematical equations.

When a set of mathematical equations from a computer simulation involve a lot of simplifications with respect to the target system, it is natural to say that they directly refer not to the target system, but rather to a system distinct from the latter that is then used to understand the target. Since this system typically only exists in thought, we are talking about a fictional model. In fact, when computer scientists describe their simulations, they often refer to point particles that collide fully elastically and so on. Such point particles clearly form a fictional model. Accordingly, at least some computer simulations involve a fictional model. The latter is of course intimately connected with the mathematical model in a simulation, because the latter describes the former, in particular, its dynamics.

Does a computer simulation also involve a material model? Hartmann's definition of simulation in terms of a process that imitates another one refers to processes in a computer hardware. One can in fact show that, in successful deterministic simulations, the dynamics of the programmed computer represents the dynamics of the target system: The computer runs through a sequence of states that each correspond to states in the target system that follow the same order (Beisbart 2014). But it stretches things a bit to say that the computer itself serves as a surrogate for the target system. The reason should be clear from our discussion of experiments: We cannot really say that the computer hardware itself is observed or investigated by simulation scientists because no information about the computer is obtained. So in what follows, our focus will be on the mathematical and the fictional model involved in a simulation.

Above, we have distinguished between the conceptual and the computational model. This distinction is orthogonal to the distinction between mathematical and fictional models behind simulations. It is clear that there are both a conceptual and a computational mathematical model depending on whether we talk about equations that the scientists are really interested in on the basis of their knowledge or whether these are approximations that have to be made to implement the former equations in the computer. Likewise, we can apply the distinction between conceptual and computational models at the level of fictional models, if the assumptions about the imagined system that are inherent in the simulation program are not exactly the assumptions that scientists have started with. In particular, very often, the dynamics of the fictional system traced by the simulation program is not exactly the dynamics of the fictional system that scientists were originally interested in. This is of course due to approximations needed for the implementation of the conceptual model in a digital computer.

Given that there are various models associated with simulations, the question arises of what the point of running the computer program is. The answer is that a run of the computer program contributes to the analysis of the model (i.e., the second stage of modeling). As far as the mathematical equations are considered to be a model, the run of the simulation program yields information about its solutions. As far as a fictional model is concerned, information about its dynamical behavior is derived by working out an (approximate) solution to the equations that describe the fictional system. In either case, if all goes well, the simulation scientist first and foremost gains knowledge about the model.

The analysis of a conceptual model (be it mathematical or fictional) can be very difficult and lead to errors. The reason is that the conceptual model is some distance away from what the computer does in fact do. The verification of a simulation thus is supposed to make a case that the analysis produces genuine results about the conceptual model. Thus, what is called verification of a simulation is the internal validation$^{\mathrm{m}}$ of results about the conceptual model.

If such results have been established, they have to be translated to the target to make some progress in knowledge about, or understanding of, the latter. In terms of our modeling terminology, we may say that we need external validation$^{\mathrm{con/com}}$ of the conceptual or computational model to make sure that what the model suggests for the target holds true of the latter with sufficient accuracy. But the values that the computational model suggests for certain characteristics of the target are just the numbers output from the simulation runs. Thus, the results from the computational model (be it fictional or mathematical) are the results from the simulation. So, to validate$^{\mathrm{cs}}$ the simulation is to validate$^{\mathrm{com}}$ the computational model, and vice versa. This is in fact what we have proposed in the introduction, where validation$^{\mathrm{cs}}$ was defined to be validation$^{\mathrm{com}}$ of the computational model. We now see the justification for this. If the results from the simulation have been verified regarding the conceptual model, then the validation$^{\mathrm{cs}}$ of the simulation will also establish the external validity$^{\mathrm{con}}$ of the conceptual model; then, with some right, this validity$^{\mathrm{con}}$ may be called the validity$^{\mathrm{cs}}$ of the simulation too.

It thus turns out that both verification and validation$^{\mathrm{cs}}$ of a simulation can be understood in the terms familiar from modeling. In particular, the validation$^{\mathrm{cs}}$ of the computer simulation turns out to be the validation$^{\mathrm{com}}$ of the computational model, when we adopt the modeling terminology. As a consequence, principles from the methodology of modeling can be used to validate computer simulations. The problem is only that a neat and tidy methodology for validating models is as much missing as one for computer simulations. When we talk about fictional models, validation$^{\mathrm{m}}$ would have to show that some results on the fictional model carry over to the target because both are similar in relevant respects. Concerning mathematical equations, the point of validation^{m} is to show that the equations provide results that are accurate enough for the purposes of a simulation. Either way, there doesn't seem any general principled approach to achieve this. So we cannot simply draw on rich insights into modeling to make progress on understanding the validation^{cs} of simulations.

37.6 Conclusions

One strategy to understand validation^{cs} of computer simulation is to begin with the question of which sort of method computer simulations are, or, maybe, how they are associated with other methods, and then to derive consequences for validation^{cs}. In this chapter, we have pursued this strategy. What did we earn by doing so?

Computer simulations can be associated with several methods that have some independent life and that have in fact been practised before the advent of computer simulation. As it happens, the term "validation" has currency regarding some of these methods too.

Although computer simulations are not running *experiments*, properly speaking, some simulations model possible experiments. An immediate consequence is that, to some extent, the validation^{cs} of experimental results need to be modeled too. But there is more to the validation^{cs} of computer simulation: It must be shown that they properly trace the target system, in particular, its reaction to an intervention. It turns out that techniques for doing so have close parallels in the methodology of experiments.

Carrying out a computer simulation is much closer to going through a *thought experiment* than running a real experiment. Both computer simulation and thought experiments do not involve causal interaction with the target system. But thought experimentation is a peculiar method; there is no worked out methodology, and the philosophical explanation of how thought experiments work (if they do) is controversial. A useful approach to at least many thought experiments is the so-called argument view. It can be extended to computer simulations and then basically cashes out the idea that computer simulations infer what a model implies for the dynamics of its target. From this perspective, the main question of validation^{cs} is whether the inference constituted by a computer simulation is sound. It is natural to split this question into two questions, viz., whether the premises are sufficiently accurate (at least as far as their impact on the results is concerned) and whether the argument is such that the premises support the conclusion sufficiently. Whether the premises are sufficiently accurate is a matter of validation^{m} of the underlying model. Whether or not the conclusion is sufficiently supported by the premises is in principle a matter of logic but cannot be investigated using techniques from logic or argumentation theory in the case of simulations. The reason is that it is not explicit how the results follow from the premises because simulations are in some sense opaque. Further, in typical examples of computer simulations, no independent case for the validity^{m} of the conceptual model can be made. Thus, the comparison between simulation outputs and measured data is the preferred method of validation. The argument view can accommodate this but doesn't have particularly interesting implications for it.

Computer simulation is finally closely related to *modeling*. Each simulation implements a mathematical model. If the mathematical model is very simplistic, when compared to the target, it is most natural to say that its equations directly refer to a fictional system that is then used to learn about the target system. So in this case, the computer simulation is closely associated with a fictional model too. Regarding both the mathematical and the fictional model, we may distinguish between the conceptual and the computational model. What the computer simulation qua run of the computer program does is to analyze a computational model. If verification is successful, the results on the computational model can be interpreted in terms of the conceptual model. Since the most interesting part of validation$^{\mathrm{m}}$ of a model establishes that the results obtained for the model can be translated to the target, validation$^{\mathrm{cs}}$ of the simulation is validation$^{\mathrm{com}}$ of the computational model. But this doesn't allow for very interesting insights about the validation$^{\mathrm{cs}}$ of simulations.

Our results are not without irony. Although computer simulations are not experiments (or so has been argued), the methodology of experiment seems to provide the most fruitful perspective on the validation$^{\mathrm{cs}}$ of simulations because many strategies of validating$^{\mathrm{cs}}$ simulations have close parallels in the methodology of experimentation (Parker 2008). After some reflection, this shouldn't come as a surprise, however. Experimentalists can draw on a long track record of successful experimentation. There is nothing like this for thought experiments; also, the arguments behind computer simulations cannot be surveyed and assessed with the techniques from argumentation theory. Modeling, finally, is too close to simulation as to allow for an interesting perspective on the validation$^{\mathrm{cs}}$ of simulations. These days, modeling and computer simulation are so much intertwined that we cannot expect that there is an independent storehouse of recipes for modeling that may then be used for the validation$^{\mathrm{cs}}$ of computer simulations. It is true that there has been, and still is, a lot of modeling without computer simulation. But modeling of this sort has often remained content with qualitative agreement with the target and is not much concerned with predictions of high accuracy, which is a vital issue for many simulations.

A possible objection against the claims of this chapter may be that it has been friendly to various accounts of computer simulation. But do they really fit together? I don't see any problems in this respect. That computer simulations can, and often do, model possible experiments, nicely fits with the view that computer simulations implement models. Now when we talk about the models implicit in simulations, we often do not say that they model a possible experiment. To some extent, this is so because some simulations (viz., those that trace a target system that is not manipulated during an experiment) do *not* in fact model experiments. For other simulations, we can say that they model an experiment but this is not absolutely necessary for their understanding. The view that computer simulations can, and do in fact sometimes do, model possible experiments is also compatible with the idea that they are something like thought experiments or arguments. The reason is that the result of the modeled experiment is inferred using the help of a computer and thus in a way anticipated in thought.

As we can fit together the various accounts of computer simulations accepted in this chapter, we can piece together the implications for the validation^{cs} of computer simulations. What the accounts suggest, for instance, are certain distinctions within the activities of validating^{cs} simulations. Clearly, several ways of drawing such a distinction can be appropriate and useful. The distinctions may further be used to propose certain strategies to validation^{cs} of simulations, and it is no contradiction to say that there are various strategies to validate^{cs} simulations.

Acknowledgements I'm grateful to Julie Jebeile and Nicole J. Saam for useful comments and criticism.

References

Baumberger, C., Knutti, R., & Hirsch Hadorn, G. (2017). Building confidence in climate model projections: An analysis of inferences from fit. *WIREs Climate Change*, *8*(3), e454.

Beisbart, C. (2012). How can computer simulations produce new knowledge? *European Journal for Philosophy of Science, 2*(3), 395–434.

Beisbart, C. (2014). Are we sims? How computer simulations represent and what this means for the simulation argument. *The Monist*, *97*(3), 399–417, (special issue edited by P. Humphreys).

Beisbart, C. (2018). Are computer simulations experiments? And if not, how are they related to each other? *European Journal for Philosophy of Science, 8*(2), 171–204. https://doi.org/10.1007/s13194-017-0181-5.

Beeler, J. R. (1983). *Radiation effects computer experiments*. Amsterdam etc: North-Holland.

Brown, J. R. (1991). *The laboratory of the mind: Thought experiments in the natural sciences*. London: Routledge.

Brown, J. R. (2004). Peeking into Plato's haeven. *Philosophy of Science,* 71, 1126 –1138.

Brown, J. R., & Fehige, Y. (2017). Thought experiments. In E. N. Zalta (Ed.), *The stanford encyclopedia of philosophy* (Summer 2017 Edition). https://plato.stanford.edu/archives/sum2017/entries/thought-experiment/.

Campbell, D. T. (1957). Factors relevant to the validity of experiments in social settings. *Psychological Bulletin, 54*(4), 297–312. https://doi.org/10.1037/h0040950.

Campbell, D. T., & Stanley, J. C. (1963). Experimental and quasi-experimental designs for research. In N. L. Gale (Ed.), *Handbook of research on teaching* (pp. 88ff). Chicago, IL: Rand McNally.

Clark, A., & Chalmers, D. (1998). The extended mind. *Analysis, 58*(1), 7–19.

Cook, T. D., & Campbell, D. T. (1979). *Quasi-experimentation: Design & analysis issues for field settings*. Boston: Houghton Mifflin Company.

Einstein, A. (1961). Relativity, the special and the general theory. A Popular Exposition. London: Methuen (1920, here quoted after edition published by Crown, New York).

El Skaf, R., & Imbert, C. (2013). Unfolding in the empirical sciences: experiments, thought experiments and computer simulations. *Synthese, 190*(16), 3451–3474.

Franklin, A., & Perovic, S. (2016). Experiment in physics. In E. N. Zalta (Ed.), *The stanford encyclopedia of philosophy* (Winter 2016 Edition). https://plato.stanford.edu/archives/win2016/entries/physics-experiment/.

Frigg, R., & Hartmann, S. (2017). Models in science. In E. N. Zalta (Ed.), *The stanford encyclopedia of philosophy* (Spring 2017 Edition). https://plato.stanford.edu/archives/spr2017/entries/models-science/.

Frigg, R. P., & Reiss, J. (2009). The philosophy of simulation: Hot new issues or same old stew? *Synthese, 169,* 593–613.

Gendler, T. S. (2004). Thought experiments rethought and reperceived. *Philosophy of Science, 71,* 1152–1163.

Gupta, A. (2015). Definitions. In E. N. Zalta (Ed.), *The stanford encyclopedia of philosophy* (Summer 2015 Edition). https://plato.stanford.edu/archives/sum2015/entries/definitions/.

Hartmann, S. (1996). The World as a process: Simulations in the natural and social sciences. In R. Hegselmann et al. (Eds.), *Modelling and simulation in the social sciences from the philosophy of science point of view, Theory and decision library* (pp. 77-100). Dordrecht: Kluwer.

Heidelberger, M. (2005). Experimentation and instrumentation. In D. Borchert (Ed.), *Encyclopedia of philosophy*. Appendix (pp. 12–20). New York: Macmillan.

Hughes, R. I. G. (1997). Models and representation. *Philosophy of Science (Proceedings), 64,* S325–S336.

Humphreys, P. (2004). *Extending ourselves: Computational science, empiricism, and scientific method*. New York: Oxford University Press.

Humphreys, P. (2009). The philosophical novelty of computer simulation methods. *Synthese, 169,* 615–626.

Imbert, C. (2017). Computer simulations and computational models in science. In L. Magnani & T. Bertolotti (Eds.), *Springer handbook of model-based science* (Vol. 34, pp. 733–779). Cham: Springer.

Koblick, D. C. (1959). An enzymatic ion exchange model for active sodium transport. *The Journal of General Physiology, 42*(3), 635–645.

Lenhard, J. (2011). Epistemologie der Iteration. *Gedankenexperimente und Simulationsexperimente. Deutsche Zeitschrift für Philosophie, 59*(1), 131–145.

Liu, J., Wang, M., Chen, S., & Robbins, M. O. (2010). Molecular simulations of electroosmotic flows in rough nanochannels. *Journal of Computational Physics, 229*(20), 7834–7847.

Massimi, M., & Bhimji, W. (2015). Computer simulations and experiments: The case of the Higgs boson. *Studies in History and Philosophy of Modern Physics, 512,* 71–81.

Morrison, M. (2009). Models, measurement and computer simulation: The changing face of experimentation. *Philosophical Studies, 143,* 33–57.

Morrison, M. (2015). Reconstructing reality: Models, mathematics, and simulations. New York: Oxford University Press.

Naumova, E. N., Gorski, J., & Naumov, Y. N. (2008). Simulation studies for a multistage dynamic process of immune memory response to influenza: Experiment in silico. *Annales Zoologici Fennici, 45,* 369–384.

Nersessian, N. J. (1992). In the Theoretician's laboratory: Thought experimenting as mental modeling. In *PSA: Proceedings of the Biennial Meeting of the Philosophy of Science Association* (Vol. 1992, pp. 291–301).

Nersessian, Nancy J. (2007). Thought experimenting as mental modeling. *Croatian Journal of Philosophy, 7*(2), 125–161.

Newton, P., & Shaw, S. (2014). *Validity in educational and psychological assessment*. London: SAGE Publications.

Norton, J. D. (1996). Are thought experiments just what you thought? *Canadian Journal of Philosophy, 26,* 333–366.

Norton, J. D. (2004a). On Thought experiments: Is there more to the argument?. In *Proceedings of the 2002 Biennial Meeting of the Philosophy of Science Association. Philosophy of Science* (Vol. 71, pp. 1139–1151).

Norton, J. D. (2004b). Why thought experiments do not transcend empiricism. In C. Hitchcock (Ed.), *Contemporary debates in the philosophy of science*. Blackwell: Oxford, pp. 44–66.

Parker, W. S. (2008). Franklin, Holmes, and the epistemology of computer simulation. *International Studies in the Philosophy of Science, 22*(2), 165–183.

Parker, W. (2009). Does matter really matter? *Computer Simulations, Experiments, and Materiality, Synthese, 169,* 483–496.

Radder, H. (2009). The philosophy of scientific experimentation: A review. *Automatic Experimentation 1*. Open access. http://www.aejournal.net/content/1/1/2.

Saam, N. J. S. (2017). What is a computer simulation? *A Review of a Passionate Debate, Journal for General Philosophy of Science, 48*(2), 293–309.

Schlesinger, S. et al. (1979). Terminology for Model Credibility, *Simulation, 32*, 103–104.

Suárez, M. (2004). An inferential conception of scientific representation. *Philosophy of Science, 71,* 767–779.

Verlet, L. (1967). Computer "experiments" on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules. *Physical Review*, *159*(1), 98.

Weisberg, M. (2007). Who is a modeler? *British Journal for Philosophy of Science, 58,* 207–233.

Winsberg, E. (2001). Simulations, models, and theories: Complex physical systems and their representations. In *Proceedings of the Philosophy of Science* (Vol. 68, pp. 442–454).

Winsberg, E. (2003). Simulated experiments: Methodology for a virtual world. *Philosophy of Science, 70,* 105–125.

Winsberg, E. (2009). A tale of two methods. *Synthese, 169,* 483–496.