**ORIGINAL RESEARCH**

# Computer simulations and surrogative reasoning for the design of new robots

**Edoardo Datteri[1] · Viola Schiaffonati[2]**

## Abstract
Computer simulations are widely used for surrogative reasoning in scientific research. They also play a crucial role in engineering, more specifically in the design of new robotic systems, yet the nature of this role has been little discussed so far in the philosophy of technology literature. The main claim made in this article is that the notion of surrogative reasoning is central to understanding how computer simulations can serve the purpose of designing new robots. More specifically, it is argued that computer simulations can support two forms of surrogative reasoning, which are called model-oriented and prediction-oriented, whose inferential structure is reconstructed to some extent. And it is argued that, when computer simulations are used to design new robots, they are distinctively used in the model-oriented way. By unravelling the structure of the computer simulation-supported methods adopted in robotic design, this article may contribute to a finer-grained understanding of the epistemic processes involved in technological research.

**Keywords** Computer simulations · Simulation-supported surrogative reasoning · Engineering design · Robotics · Robotic models

## 1 Introduction

It has long been recognized that engineering involves stages of construction and understanding. To refine an existing artefact, one must understand how it works and theorise on what should be changed to improve its functionality. To design and build a brand-new artefact, one must predict how it would behave if it was designed in a certain

✉ Viola Schiaffonati
viola.schiaffonati@polimi.it

1 RobotiCSS Lab, Laboratory of Robotics for the Cognitive and Social Sciences, Department of Human Sciences for Education, Università degli Studi di Milano-Bicocca, Piazza dell'Ateneo Nuovo 1, 20126 Milan, Italy

2 Artificial Intelligence and Robotics Laboratory, Department of Electronics, Information, and Bioengineering, Politecnico Di Milano, Piazza Leonardo da Vinci 32, 20133 Milan, Italy

way or another. The importance of understanding and prediction in engineering is acknowledged and extensively discussed by van Eck (2016). Yaghmaie (2021, p. 456) points out that "engineers before everything else are going to predict the properties of a to-be-built artifact, to explain its future mechanism and to understand its prospective behaviour". Notably, as pointed out by Boon and Knuuttila (2009), *models* of the (existing or to-be-built) system are generally used to support or achieve these epistemic purposes. This consideration gives rise to several philosophical problems that are akin, or connected, to philosophical problems addressed in the philosophy of model-based science. One is the so-called constitutive problem of engineering models: "in virtue of what does an engineering model model its target system?" (Yaghmaie, 2021, p. 455). It is not obvious that the relationship holding between models and target systems in scientific research is the very same relationship holding between models and target systems in engineering. More specifically, one may question whether the representation relation characterising model-based science differs from the "design relation" holding between models and their targets in engineering, which is a problem discussed by Poznic (2016). Another philosophical problem arising in connection to the use of models in engineering is the so-called "problem of the non-existing artifact". Models of systems that are yet to be built are targetless. How can people utter and communicate truths about things that are not there to make their propositions true? This problem has been discussed by Galle (1999).

This article offers an analysis of the structure of some model-based epistemic processes underlying the design of particular kinds of objects, viz., robotic systems. It specifically deals with the case in which computer simulations are used *qua* concrete models[1] as tools to assist in designing new robots. The main claim made here is that one of their possible roles is to support a particular form of surrogative reasoning (on the robot that is to be built) that is called "model-oriented, simulation-supported surrogative reasoning", and labelled MO SsSR.[2] This claim will be supported by analysing the notions of computer simulation, surrogative reasoning and model-oriented simulation-supported surrogative reasoning, and by showing—also by reference to an example—that some simulation-supported design processes in robotics take a model-oriented structure.

This work is meant to contribute to the philosophical debate on simulations in two distinct ways. First, it should complement the existing literature on the structure of engineering design processes. van Eck (2016) points out that engineering theoretical models—that, in his framework, decompose the system to be built into an organised set of functional components—are primarily used in design as a means to formulate counterfactual predictions of the form, i.e., "what would happen if things were different", e.g., what would happen if one functional component was replaced by a different one. Here it is pointed out that computer simulations, *qua* physical systems

---

[1] The distinction between concrete and non-concrete models has often been made in the philosophical literature, for example, by Frigg and Nguyen (2017), whilst Weisberg (2013) distinguishes between concrete, computational, and mathematical models. Here, concrete models are taken to be those models which admit a physical description.

[2] It is not claimed here that this is the only possible role for computer simulations in the design of new robots.

implementing those theoretical models, may be used to formulate predictions, possibly counterfactual in van Eck's sense, on the system to be built (here these will be called *conditional predictions*).

Therefore, this article discusses one way in which the epistemic role of engineering theoretical models envisaged by van Eck can be practically fulfilled. At the same time, by offering a reconstruction of the structure of computer simulations, this paper is meant to pave the way for reflection on the factors that can threaten the validity of those predictions. Second, this research aims to enrich the philosophical literature on computer simulations, by showing that they may play an important epistemic role in engineering processes too (whereas the philosophical literature has been mostly focused on the epistemic role of computer simulations in "pure" sciences) (Datteri and Schiaffonati 2019).

More specifically, Sect. 2.1 will be devoted to an analysis of the notion of a computer simulation. Drawing from the contemporary philosophical literature on the topic, the relationship between concrete simulation systems, simulation models, and theoretical models of the modelled systems will be outlined. Section 2.2 will offer a characterisation of the notion of surrogative reasoning, which shows that there are two forms called prediction-oriented and model-oriented surrogative reasoning that can be distinguished. Section 3 will bring these considerations to bear on the design of new robots. In Sect. 3.1 the structure of design processes in robotics will be sketched out, drawing from the literature on the philosophy of technology. In Sect. 3.2 it will be shown that some parts of these processes may involve computer simulations that are used *qua* concrete models to support MO SsSR on the robot that is to be built. This claim will be exemplified with reference to a study on the control of the whole-body dynamic behaviour of human-like robots.

## 2 Theoretical background

### 2.1 Simulation

The term "simulation" takes on several meanings in science and philosophy. According to Humphreys (1990), simulations are *methods* for exploring the properties of mathematical models where analytic methods are unavailable. Ord-Smith and Stephenson (1975) define simulations as *techniques* that involve observing a model to understand a physical system. Hartmann (1996) takes simulation to denote the *result* of solving the equation of a dynamic model using a computer. The term may be used to denote an algorithm (as in Winsberg, 2009), an activity, a process, or a concrete system (e.g., "this computer is a simulation of the Lotka-Volterra equations"). This list is not exhaustive: other usages can be found in the literature (see Imbert, 2017 for a comprehensive overview). To avoid terminological misunderstandings, here the terms "simulation system" and "simulation" are used with different meanings. "Simulation system" is used to denote the physical system that carries out the simulation. In computer simulations, the simulation system is a digital computer considered as a physical system. Depending on the context, "simulation" is instead used here to denote the act of simulating, the process carried out by a simulation system, or, in the broad sense

introduced by Frigg and Reiss (2009), the entire process of building a simulation system and using it to achieve some scientifically interesting goal.

As more extensively discussed in the next section (2.2), simulation systems are typically used as surrogates to learn something about other systems. It is assumed that, when a simulation system is used for this purpose, it can be regarded as a concrete model of the system under investigation, which is called the "target system" here in coherence with the philosophical literature on models.[3] The analysis offered in this paper rests on the assumption that the target system may not exist when the simulation is carried out: one may simulate the behaviour that a non-existing object would have under certain conditions (this is a possible use of simulations in robotics design, as clarified below). This gives rise to the difficult problem of understanding what kind of relationship holds between simulation systems and target systems that do not exist. This problem has been extensively studied in the philosophical literature[4] and will not be addressed here, as it is beyond the scope of the article: it will be simply assumed that simulation systems can be models of non-existing systems and that the relationship between the two can be characterised in some way.

To understand the specific epistemic role played by computer simulations in the design of new robots it would be helpful to reconstruct the relationship between simulation systems, *qua* models, and their targets. Here, borrowing from the relevant literature,[5] it will be assumed that any given simulation system realises a simulation model that implements a discretized version of a theoretical model of a target system plus several integration modules. In some cases, the simulation model also comprises modules simulating the environment in which the target system is (expected to be) located. This section elaborates on the details of this structure, which is depicted in Fig. 1.

The simulation system, *qua* physical computer, realises the simulation model, which is a computer program, typically modular and complex.[6] The simulation model implements a discretized version of a theoretical model of the target system. In some

---

[3] Justifying this assumption would require one to evaluate the relationship between computer simulations and their target systems against some philosophical account of what makes an object a (concrete) model of another object (a detailed survey can be found in Frigg & Nguyen, 2017). This goal will not be pursued here. In this respect, this article aligns with some of the most influential philosophical accounts of computer simulations (including Weisberg, 2013; Winsberg, 2003), which regard computer simulations as models.

[4] Consider, for example, Frigg and Nguyen's (2017) discussion of the problem of targetless models, which they introduce as follows: "Models of ether, phlogiston, four-sex populations, and so on, are all deemed scientific models, but ether, phlogiston, and four-sex populations don't exist. Such models lack (actual) target systems, and one hopes that an account of epistemic representation would allow us to understand how these models work" (p. 54). One particular version of the problem of the targetless models is the problem of the absent artefact (Galle, 1999), which calls for an understanding of how engineering models can be used to support truths about non-existent entities. So-called fictional accounts of scientific models are particularly suited to accommodate the possibility of targetless models (Frigg, 2010). Massimi (2019) shows how targetless models successfully deliver varieties of modal knowledge in science.

[5] This reconstruction is not meant to add anything novel to the literature on computer simulations and simulation models. In the form presented here, it is largely consonant with (Durán, 2020). However, it is not claimed here that this is the only framework functional to the purposes of this paper.

[6] The nature of the realisation relationship that may hold between the simulation model and the simulation system—more generally, the problem of determining under what conditions a physical system can be said to realise a given program—is beyond the scope of this article (see Rescorla, 2014 for a proposal).
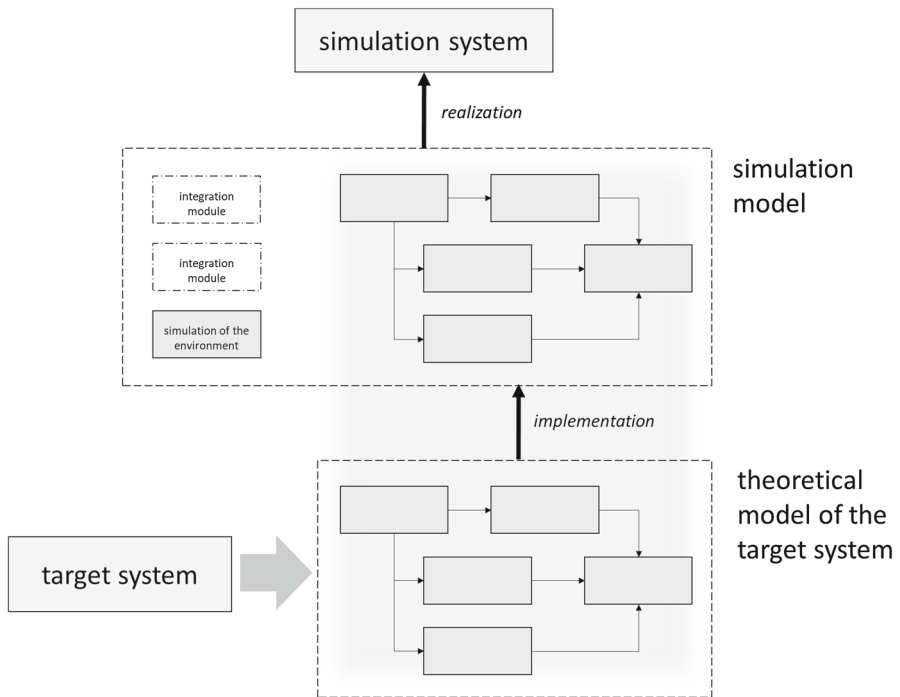
**Fig. 1** The relationships between target system, theoretical model, simulation model, simulation system

cases—including those covered by the discussion made in Sect. 3—the theoretical model takes the form of a functional (or mechanistic) model,[7] decomposing the target system into an organized set of functional modules $T_1$, …, $T_n$. Therefore, the distinction between a simulation model and a theoretical model needs to be clarified further. The simulation model is the program "running" on the simulation system, whilst the theoretical model is the model of the target system that the researcher uses when theorising on it. Even though, in principle, the theoretical model may coincide with the simulation model, in some cases it will not. Robotics is a case in point. Simulation models of robotic systems typically do not coincide with theoretical models of those systems, especially because (1) robots are not fully algorithmic systems, and (2) simulation models of robots typically include components that are unrelated to the simulated robot.

---

[7] Functional models are taken here to be theoretical models that decompose the target system into a set of functional modules, i.e., parts of the target system identified by the function they play in the framework of the system. For example, a functional model of a robotic system may mention a "distance sensor" module whose function is to acquire information on the distance between the robot and the closest object. For some purposes, it may be important to characterise functional modules also in terms of input-output regularities. It is also assumed here that functional models may describe mechanisms—in other words, that mechanisms can be described functionally, as organised sets of functionally identified parts. This working definition suffices for the present purposes, and a no more specific account of functional modelling will be provided here.

As far as point (1) is concerned, recall that the simulation model is a program. Thus, if the simulation model is regarded as a theoretical model of the target system, then the researcher *uses a program as a theoretical model* to theorise about the target system. In some areas of scientific research, chiefly including cognitive science, it has been claimed that programs can be directly regarded as theories (versions of this idea have been put forth by Cummins, 1977, 1983; Johnson-Laird, 1983). However, computer simulations are carried out also in areas that do not adopt this epistemological approach. If the simulation model is a program, and the theoretical model is not couched in that format, then the simulation model will be conceptually distinguishable from the theoretical model. Neuroscience and physics—fields in which computer simulations are frequently used—are cases in point. In engineering, as more extensively discussed in Sect. 3, the target system's theoretical model typically consists of a functional model, which is *prima facie* different from a program. One may object that a functional model can be conceived as a modular program, the function of each module being fulfilled by a part of the program. However, leaving aside considerations of abstraction and generality (functional models typically do not specify the details of the program running "inside the boxes"), there is a good reason to believe that theoretical (functional) models of *robotic* systems, to be useful for epistemic (predictive, explanatory) and constructive purposes, cannot be couched in terms of programs. Robots are composed of sensors, effectors, and a control system. Typically, sensors and effectors are modelled in non-algorithmic terms. For example, direct current motors are generally modelled in terms of a relationship between electrical energy and mechanical energy, where the "input" and "output" parameters range over the set of real numbers. Some sensors (e.g., photoresistors) are modelled in terms of relationships between properties of the environment and electrical properties of the component itself, both ranging over the set of real numbers. Their input–output behaviour can be simulated using a program, but one thing is that program, and another thing is the functional representation that is used as a blueprint to build the robotic system (e.g., to choose the "right" physical sensor on the market) or as a theoretical basis to formulate predictions or explanations.

The control system needs a separate consideration. Most robots are controlled by a digital computer on which a program is running. Thus, this particular component of the target system—namely, the control system—is a program "in itself". More precisely, for most epistemic and constructive purposes, the theoretical model of a robotic system will consist in a combination of a non-algorithmic functional model (the sensors, the effectors) and a functional model that is specified algorithmically (the control system). The control module is a zone of (partial[8]) overlapping between the theoretical model and the simulation model.

There is another reason for distinguishing between the two. As pointed out before, the simulation model is a program that must run on a digital computer. This implies

---

[8] The overlap may only be partial. It can be that the theoretical model of the control system is a fully specified program. In this case, there may be a full overlap (if the very same program "runs" in the simulation system too). Or it could be that the theoretical model of the control system is an algorithm specified at a higher level of abstraction or containing gaps—for example, the designer might be content, given their current purposes, with a sketchy and abstract description of the control system. In this case, the corresponding module of the simulation model will not coincide with that description.

that the simulation model will have to include modules that are needed for it to run correctly, even though they are unrelated to the theoretical model. These include what Durán ([2020]) calls *integration modules*, which integrate external databases, protocols, libraries and the like with the rest of the program, and ensure synchronisation, efficient data exchange, and internal compatibility. Moreover, in simulations of robots, the simulation model typically includes modules that simulate the environment in which the simulated robot operates. The fact that the simulation model includes a variety of modules that are neither related to, nor constrained by, the theoretical model of the target system is another reason for believing that the two kinds of models must be kept conceptually distinct from one another.

The considerations made so far presuppose neither that, in the building of a simulation system, one always formulates the theoretical model temporally *before* the implementation of the simulation model, nor that one always must make the theoretical model explicit. One may write the program first and *then* derive the theoretical model of the target system. Or one may simply be content with writing the program and observing its outcomes. So, what is the point of distinguishing the simulation model from the theoretical model, and claiming that the former implements a discretization of the latter? The role of the theoretical model clearly emerges when the simulation system is used in the framework of an engineering (robotic) construction process. In engineering, as more thoroughly discussed below, the simulation system is used to test a design of the system to be built. Based on the simulation outputs, one may be led to accept or reject a particular design—i.e., to conclude that the design under scrutiny can be sent to production, or that it fails and should be rejected as it stands. For the reasons illustrated above, the simulation model does not display the "right" features for being a design of the robotic system. The simulation model is a program, while some robotic components are non-algorithmic in the sense clarified above. Moreover, it contains parts that are totally unrelated to the system to be built. Therefore, it is the theoretical model that plays a crucial role in the design and construction process. Though the designer may well start from a simulation model, without dealing first with the complexities of formulating a theoretical model of the robot to be built, when it comes time to build the system, they will have to "translate" the simulation model back into a theoretical model which has fewer (and partially different) modules, some of which will be non-algorithmic. The text of a computer program simulating a mechanical arm is not particularly helpful as a blueprint to decide what mechanical and electrical properties a real-life robotic arm must have to fulfil the intended role.

## 2.2 Two kinds of simulation-based surrogative reasoning

"Surrogative reasoning" is a term introduced by Swoyer ([1991]) and largely used in the literature about scientific modelling. Surrogative reasoning, in Swoyer's terms, consists in "reason[ing] directly about a representation in order to draw conclusions about the things that it represents" (p. 449). Surrogative reasoning, defined in these terms, is one of the most widespread uses of models *qua* representations of other systems. Indeed, Frigg and Nguyen ([2017]) claim that every acceptable theory of scientific representation must account for how models can be used as surrogates for reasoning about

their target systems. In Sect. 2.1, a schematic analysis of the structure of computer simulations was offered, according to which any given simulation system realises a simulation model implementing a discretized version of a theoretical model of a target system. The purpose of this section is to suggest that simulation systems can be used to support two particular forms of surrogative reasoning called here "prediction-oriented" and "model-oriented". Surrogative reasoning performed via computer simulations, or simulation-supported surrogative reasoning, will be referred to as SsSR for short. Note that it is not claimed here that these two forms of surrogative reasoning can be supported by computer simulations only: indeed, there are good reasons for believing that they can be carried out using other kinds of models too, and in other branches of scientific research. The goal of this article is to reflect on how simulation systems can be used *within* the process of designing new robots, and not to identify forms of surrogative reasoning that are specifically enabled by computer simulations. The distinction between prediction-oriented and model-oriented SsSR is made here in order to single out the particular form of surrogative reasoning that computer simulations support in the design of new robots—viz., as later explained, the model-oriented one.

Some remarks on SsSR will be helpful to elaborate on this point. First, SsSR is conceived here as an *activity* carried out by a human agent. It will therefore be said that, in SsSR, an agent uses a simulation system to perform surrogative reasoning about a target system. That activity crucially involves analysing the behaviour of the simulation system in particular circumstances. Second, SsSR is taken here to be an activity that involves *drawing conclusions about the target system* based on the behaviour of the simulation system—in other words, an activity that leads the agent to accept or reject (more generally, to test) a hypothesis about the target system.[9] Putting these things together, SsSR is conceived here as an activity in which an agent tests (i.e., accepts or rejects) a hypothesis about the target system based on the behaviour of the simulation system.

SsSR may lead the agent to accept or reject various kinds of hypotheses about the target system. In some cases, SsSR is performed to learn something about the behaviour that the target system would display when some conditions C occur. A neuroscientific example is discussed in (Datteri, 2020): Reimann and colleagues (2013) use a large-scale simulation system to learn how particular brain signals, called local field potentials, would vary under particular input and boundary circumstances. The hypothesis that the agent accepts or rejects, in this case, may be reconstructed using the following template: "in conditions C, the behaviour of the target system would be such and such", where "such and such" stands for a behavioural description. Meteorological simulations used to predict tomorrow's weather are another case in point: when the agent accepts the hypothesis that tomorrow it will rain, based on the output of the simulation system, they accept a hypothesis stating that in some temporal circumstances (i.e., tomorrow) the atmospheric system will display a certain behaviour.

---

[9] "Accepting" and "rejecting" a hypothesis consist in including that hypothesis in, or excluding it from, the set of hypotheses that are believed to be true by the model user. The term "hypothesis" is intentionally used here in a very general sense: a hypothesis about the target system is any statement concerning the target system. The hypotheses considered in this paper may take the form of behavioural predictions (about the target system) or of descriptions of mechanisms (governing the behaviour of the target system).

The term "prediction" is often used in the philosophical literature to denote statements concerning the behaviour that a system would display in some circumstances, not necessarily occurring in the future: for this reason, this kind of SsSR will be called "prediction-oriented" (PO) from now on (and it will be said that, in this case, the agent is making a PO use of the simulation system). To sum up, the working definition of PO SsSR that will be used from now on is as follows.

> PO SsSR: an activity in which an agent tests a hypothesis of the form "in conditions C, the behaviour of the target system would be such and such" based on the behaviour of a simulation system that realises a simulation model implementing a discretized version of a theoretical model of the target system.

In other cases, SsSR is performed to test a hypothesis about the *mechanism* producing the behaviour of the target system. The hypothesis that the user accepts or rejects, in this case, has the form "the behaviour of the target system in conditions C can be produced by mechanism M", where M is the description of a mechanism. As discussed by Datteri (2020), a large-scale brain simulation was used by Hay and colleagues (2011) to test hypotheses on the mechanism producing the electrical profile of some pyramidal neurons. Meteorological simulations are occasionally used to test hypotheses on the atmospheric mechanisms that generated (or could generate) a typhoon under some circumstances. When a simulation system is used to test hypotheses of this kind, the SsSR will be called "model-oriented" (MO), and it will be said that the user is making a MO use of the simulation system. The term "mechanism description" is taken here to be akin to the notion of "theoretical model", as defined in Sect. 2.1: it is assumed that mechanisms can be described as functional models decomposing the target system into an organized set of functional modules (see also footnote 7). To sum up, the working definition of MO SsSR that will be used from now on is as follows.

> MO SsSR: an activity in which an agent tests a hypothesis of the form "the behaviour of the target system in conditions C can be produced by mechanism M", where M is the description of a mechanism, based on the behaviour of a simulation system that realises a simulation model implementing a discretized version of M.

PO and MO SsSR, conceived as activities, differ from one another in the form of the hypothesis that the user tests using the simulation system. To use a simulation system to test a hypothesis about the behaviour that the target system would display under certain circumstances is one thing, to use it to test a hypothesis about the mechanism that could produce that behaviour is another thing. The two forms of SsSR may also differ from one another in the structure of the reasoning process carried out by the user, and in the nature of the auxiliary assumptions that are needed to draw the two kinds of theoretical conclusions.

The reasoning process carried out in MO SsSR is typically comparative. In standard cases of MO SsSR, the agent compares the behaviour of the simulation system with the behaviour of the target system under specific circumstances. If the two match with one another (i.e., if the simulation system reproduces the behaviour of the target system) at some level of approximation, the user may be induced to accept the hypothesis that

M—a discretized version of which is implemented in the simulation model realised by the simulation system—could produce the behaviour under investigation. Otherwise, the agent may be induced to exclude it from the space of the possible mechanisms. This is the reasoning behind instantiations of the so-called "synthetic method" exemplified by the study by Hay and colleagues (2011) and extensively applied in Artificial Intelligence (see, for example, Simon & Newell, 1962): in order to test whether a particular mechanism is responsible for some behaviour, one implements that mechanism in a machine and assesses whether the machine is able to reproduce that behaviour.

The considerations made in the previous section enable one to acknowledge the argumentative efforts needed for the agent to justify their acceptance of that hypothesis and the factors that may threaten the validity of this inference. One has to assume that the mechanism description has been correctly "translated" into the simulation model, under some philosophical account of "correct translation", also considering the fact this translation may have involved adjustments and simplifications to ensure computational tractability. One must also assume that all the other parts of the simulation model that are unrelated to the mechanism description (notably, the integration modules) did not introduce behavioural perturbations, or find a proper way to neutralise their effects. It is also necessary to assume that the environment was accurately simulated. Moreover, many different mechanism descriptions could produce the same behaviour in simulation. MO SsSR may enable one to conclude that M *can* produce the behaviour under investigation, but additional assumptions are needed to draw the further conclusion that M describes the mechanism that *actually* governs the target system.

Such a comparative strategy is typically not performed in PO SsSR. In this kind of study, one typically analyses the behaviour of the simulation system and interprets it as the behaviour that the target system would generate under similar circumstances (using an interpretive framework that includes a mapping from the behaviour of the simulation system to the behaviour of the target system, possibly based on the analytical interpretation discussed by Contessa, 2007). As in MO SsSR, justifying one's acceptance of a PO hypothesis may require the acceptance of auxiliary assumptions that are far from self-evident. Often, the predictive validity of the simulation system is justified by arguing that it realises an accurate implementation of a "good" theoretical model of the target system. This requires one to argue, at least, that the simulation model accurately implements (whatever "accurately" may mean) a discretized version of a "good" theoretical model of the target system (whatever "good" means).[10]

One may doubt that MO and PO SsSR really differ from one another. In PO SsSR, the simulation system is used to predict the behaviour that the target system would generate under particular circumstances. But the MO SsSR may be seen as involving a prediction too. Indeed, the behaviour of the simulation system can be regarded as the behaviour that the target system would generate under particular conditions if it was governed by the hypothesized mechanism (under the auxiliary assumptions described above). The reasoning behind MO SsSR requires that the user accept this kind of prediction about the target system. The agent ascertains that the simulation system

---

[10] Note that PO and MO SsSR are described here as two kinds of reasoning activities that the user may perform several times in the context of any given study. The same simulation system can be used for PO and MO purposes at different times. See (Datteri, 2020) for a discussion.

can reproduce the behaviour of the target system, and from this consideration, they conclude that the hypothesized mechanism can produce it. To reach this conclusion, the agent must accept the "internal" hypothesis that the behaviour of the simulation system predicts the behaviour that the target system would display if it was governed by the hypothesized mechanism—and, since this prediction ended up being true, they conclude that the target system can be governed by that mechanism. To sum up, in a sense, both PO and MO SsSR involve predicting the behaviour of the target system.

On a closer look, however, the form of the two kinds of prediction is different in a sense that reflects the fundamental difference between PO and MO SsSR. While, in the PO case, one ends up accepting a hypothesis of the form "in conditions C, the behaviour of the target system would be such and such", in MO SsSR one accepts a hypothesis stating that "in conditions C, the behaviour of the target system would be such and such *if the target system was governed by the hypothesized mechanism*". To predict that tomorrow it will rain is one thing, to predict that tomorrow it *would* rain *if* a certain atmospheric theoretical model was true is another thing. To predict how some neural cells would behave under the effects of some drugs is one thing, to predict how some neural cells would behave under the effects of some drugs *if* a certain theoretical model of those cells was true is another thing. To emphasise this difference, which is crucial to the ensuing discussion, the term "conditional prediction" will be used to denote the kind of predictions made in MO SsSR, while the term "unconditional prediction" will be used to denote the predictions made in PO SsSR. The notion of conditional prediction plays a central role in the ensuing methodological analysis of simulation-based surrogative reasoning in robotics: in the next section, it will be argued that simulation systems are used for surrogative reasoning in the design of robotic systems and that, more specifically, they are used to formulate conditional predictions about the system to be built in the framework of a MO form of SsSR.[11]

## 3 Simulation-based surrogative reasoning in robotics

### 3.1 The engineering design process

Once dubbed "fake robotics" (Amigoni & Schiaffonati, 2017), computer simulations are now being recognized as tools to serve a variety of purposes in robotics. According to a recent review (Choi et al., 2021), they can be useful to simulate the dynamics of existing robotic systems and the world in which they are expected to operate. Real-world contexts are typically unstructured and relatively chaotic, especially if they are inhabited by humans. Nevertheless, current technologies enable one to simulate them to a reasonable degree of accuracy, and concurrently predict the behaviour of (simulated) existing robots in many foreseeable circumstances. These predictions may be taken as a basis to test the functionalities of the robot, in a way that is much safer and quicker as compared to real-world experiments (Collins et al., 2021). Notwithstanding

---

[11] MO SsSR involves formulating conditional predictions of this sort. This does not mean that MO SsSR *reduces to* formulating conditional predictions: as discussed in the text, there is more to MO SsSR than the formulation of a conditional prediction. MO SsSR also involves a comparison between the conditional prediction and the behaviour of the target system.

some limitations (discussed in Choi et al., 2021), computer simulations are often thought to be ways to accelerate the engineering design cycle and reduce its costs (Liu & Negrut, 2021).

Simulations are also regarded as tools for designing the robots of the future (Žlajpah, 2008), more specifically, as tools for designing and building new (i.e., not existing) robots. The considerations offered in Sect. 2 will be used from now on to argue that, when they are involved in the design of new robots, simulations are used to carry out model-oriented (MO) surrogative reasoning. Simulations therefore play, in the design of a new robot, a somewhat analogous role played in simulation-supported, model-oriented investigations on the mechanisms underlying physical, biological, and cognitive phenomena (exemplified by the aforementioned study by Hay et al., 2011): they lead one to test hypotheses on the mechanism that can produce the behaviour of the robot to be built—more precisely, its *desired* behaviour.

The key stages of engineering design processes, as recently reconstructed in the philosophy of technology and engineering literature (see Michelfelder & Doorn, 2021 for a comprehensive overview of the topic), can be sketched as follows. Engineering design starts with the definition of the behaviour of the artefact to be built. This is a complex process that has been discussed in a systematic way using the notion of "technical artefact". As defined by Vermaas and colleagues (2011, p. 7), "a technical artefact [is] a physical object with a technical function and use plan designed and made by human beings". Accordingly, technical artefacts can be characterised in terms of their *technical function* (what is the technical artefact for?), their *physical composition* (what does the technical artefact consist of?) and a *use plan* (how must the technical artefact be used?). Engineering design, regarded as the making of a technical artefact, is a process composed of different phases, in which certain function descriptions are translated into a blueprint for an artefact or a service that can fulfil these functions (Kroes, 2021). In the process of designing technical artefacts, two kinds of descriptions play a role: the functional description (the technical artefact *x* is for *y*, where *y* is an activity) and the structural one (the technical artefact *x* has this shape, mass, form, etc.). Schematically, engineering design can be seen as a process that translates functions into structures, so that the technical artefact realises the technical function, together with some specific wishes usually expressed by a client. A technical function can be interpreted as a desired physical property or capacity of the technical artefact (Vermaas et al., 2011). For example, the technical function of a vacuum cleaner robot is to autonomously clean a space.

The starting point of this process consists in formulating the wishes or desired behaviours that will later be translated into design specifications. *Design specifications* express the desired behaviour of the technical artefact. For example, a client (be it a company or a person) may wish to have a vacuum cleaner robot that is fast and cheap. The technical function (i.e., the robot vacuum cleaner's ability to clean) is a necessary but not sufficient component of the desired behaviour: for example, one might also want the vacuum cleaner to be fast and cheap. The desired behaviour of the artefact, so constructed, can be translated into different design specifications that will later constrain the actual building of the object. For instance, the cheapness wish can be translated into design specifications that prescribe one to use low-cost material or minimise the number of the components. Franssen (2020) points out that

design specifications, expressing the desired behaviour of the artefact, may include *requirements* (e.g., cheapness and quickness) as well as *constraints* that have their origin elsewhere, such as the properties of the materials or government regulations and industry standards.

Usually, clients do not state their wishes in a sufficiently detailed and precise way, so engineers intervene in this phase to refine and complete the clients' wish list. For example, the vacuum cleaner robot, in addition to being fast and cheap, must also be safe, a design specification usually taken for granted and not explicitly stated by the users. The process leading to a well-defined engineering problem involves not only detailing the wishes, but also rephrasing them in engineering terms—often, in a precise and quantitative way. For example, the "quickness" wish may be formulated in terms of the time needed for the robot to map and clean a determined portion of an environment. Moreover, iterative changes and adjustments may be needed to arrive at a stable set of design specifications. When this goal is achieved, one has what is usually called, in the literature on philosophy of technology, a *well-defined engineering problem*.

The successive stage consists in formulating a set of *designs* that may enable the artefact to meet the design specifications. Designs may be regarded as descriptions of the functional, physical, or computational structure of the system to be built (Kroes, 2021): if the design specifications state what the technical artefact should do, the design states how it will do that. Usually, designers will come out with several possible designs that need to be tested. It is at this point that computer simulations may enter the stage.

### 3.2 Computer simulations in the design of robotic systems

According to van de Poel and Royakkers (2011, p. 166), simulation is "the stage of the design process in which the designer or the team checks through calculations, tests, and simulations whether the concept designs [designs in our terms] meet the design requirements [design specifications in our terms]". How can the role of simulations be characterised, and what use is made of their output? It is argued here that, in the design of new robotic systems, computer simulations are used to support MO surrogative reasoning. In this context, computer simulations are used to discover the "right" design, i.e., the design that may enable the to-be-built robot to meet the design specifications.

The concept of "design", that is traditionally used in the philosophy of technology and engineering literature, shares some similarities with the concept of theoretical model (of the to-be-built robotic system) discussed in Sect. 2. In engineering, theoretical models of the target system are used both to predict and explain the behaviour of the target system (be it existing or non-existing), and to design and build the target system (when non-existing): in this second case, they constrain the construction process. As discussed in (van Eck, 2015), they often assume the form of functional models describing the mechanism (supposedly) governing the artefact. Functional models are models that decompose the target system into an organised set of functional modules, each one identified by the function it performs within the system. In computer simulations of robotic systems, the system modelled by a theoretical model is a robot. Each

functional module $T_1$, ..., $T_n$ of a theoretical model represents a part $t_1$, ..., $t_n$ of the target robotic system (including the sensors, the effectors, the control system or parts thereof). In this perspective, the design of a robot is the description of an organised structure of functional modules $T_1$, ..., $T_n$ representing parts $t_1$, ..., $t_n$ of the target robotic system. Designing a robotic system amounts to formulating a theoretical model of it, and a "good" design will make the system comply with the previously formulated design specifications. In this respect, design—considered as a process—is iterative: one tentatively proposes a design, tests if it lives up to its expectations, and if the test fails a revised design is produced. How can the testing be carried out in engineering design? One option is to build a concrete prototype of the robotic system, based on the theoretical model under scrutiny, and check if it meets the design specifications. However, this process may be expensive and time-consuming. Another option is to build a simulation system based on the tentative theoretical model and check if it meets the design specifications. In that case, one may be led to accept the hypothesis that the theoretical model is the "right" design; otherwise, one may be induced to reject this hypothesis and, possibly, revise the theoretical model.

Since the simulation system is used to *learn* if the robot, when built as described in the theoretical model, would meet the design specifications, it is reasonable to claim that it plays an epistemic role in the design process. More specifically, in the reconstruction of the simulation-based robotic design made here, simulation systems are used to formulate what have been called conditional predictions about the robot to be built, viz. predictions of the form "in conditions C, the behaviour of the robot would be such and such if the robot was governed by that theoretical model". In the design of a robotic vacuum cleaner, the simulation system would enable one to generate the behaviour that the robot would display if it were governed by the design under scrutiny—thus enabling one to understand whether it could meet the design specifications (e.g., whether it could clean a given portion of space in the desired amount of time). Conditional predictions, when evaluated against the design specifications, enable one to accept or reject the hypothesis that the simulated mechanism can produce that behaviour. This process is a slightly adjusted version of what has been called model-oriented simulation-supported surrogative reasoning (MO SsSR). As discussed before, MO SsSR provides evidence to accept or reject the hypothesis that a particular mechanism can produce the behaviour of the target system and can therefore play an important role in mechanism discovery. In the context discussed here—the design of a non-existing robot—the behaviour of the target system (i.e., the robot to be built) is clearly unavailable. However, the designer has what have been called design specifications, which, in the sense discussed before, describe the desired behaviour of the technical artefact. The conditional predictions about the non-existing robot generated via the simulation system are compared with its desired behaviour, and the outcome of the comparison is taken as an empirical basis to accept or reject the hypothesis that the theoretical model is a mechanism producing the desired behaviour of the robot.

To sum up. In scientific research, MO SsSR is a form of surrogative reasoning that enables one to test hypotheses concerning the mechanism that produces the known behaviour of existing systems. A key stage in robotic design consists in testing hypotheses concerning the mechanism that will produce the *desired* behaviour of a *non-existing*

system. Simulation systems, when used in robotic design, constitute surrogates for reasoning about the behaviour of the non-existing robot. As in MO SsSR, they are used to obtain conditional predictions about its behaviour. As in MO SsSR, these conditional predictions are compared with the desired behaviour of the target system, in order to assess whether a certain theoretical model could produce that behaviour. As in MO SsSR, the simulation system plays a key role in discovering a mechanism: indeed, the robot design process can be seen as the iterative discovery of one of the possible "right" mechanisms, i.e., a mechanism that, when built into the robotic system, will make it meet the design specifications (Fig. 2).

These considerations can be illustrated by referring to the study described in (Kathib et al., 2004). The authors elaborate a theoretical framework for controlling the whole-body dynamic behaviour of human-like robots and test it using computer simulation. The design of humanoid robots poses important challenges: in particular, if one wants the robot to be socially accepted and to smoothly interact with human users, its movements should be fluid and human-like regardless of how many tasks the robot is carrying out at the same time. This means that some aspects that are less important in industrial robots must be part of the *desiderata* of social humanoid robots, chiefly including smooth whole-body coordination and the ability to perform several tasks simultaneously. For example, the robot should be able to reach particular positions
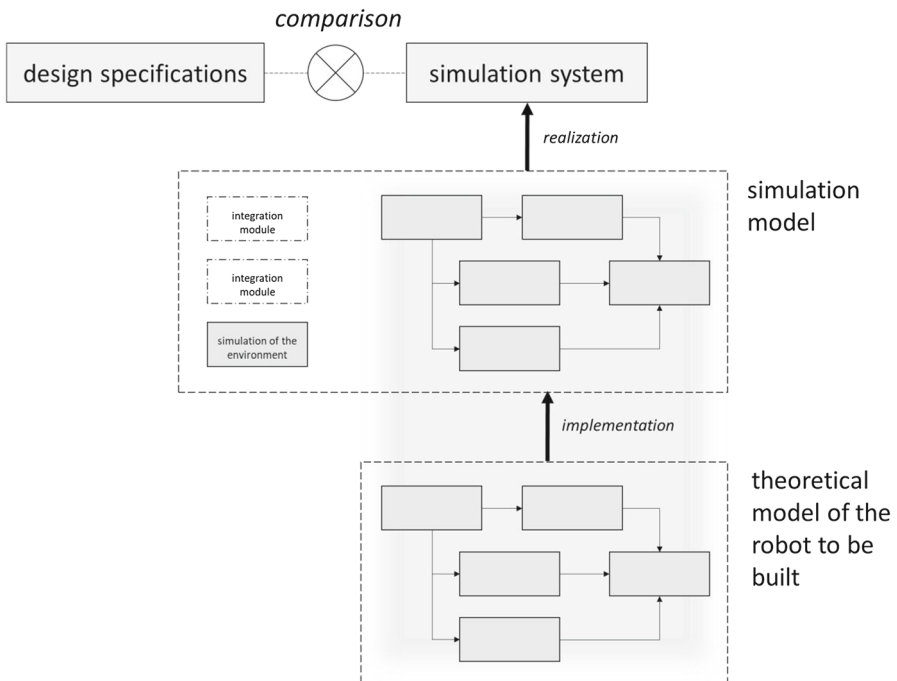


**Fig. 2** In robotic design, the output of the simulation system is compared with the design specifications. The result of the comparison may be used to test the theoretical model (or design) of the robot to be built, but may also be affected by the structure and content of the simulation model

in space with its hands, and at the same time control its global centre of mass in order to maintain whole body balance. Khatib and colleagues propose a theoretical framework that enables the robot to dynamically coordinate two tasks: a primary task (e.g., moving the hands towards a particular point) and a secondary task (e.g., postural control).

Discussing the details of this framework would go beyond the scope of this article. What matters for the present purpose is that the process of designing this framework can be reconstructed using the terminology introduced here. The *wish* consists in the development of robots whose movements are as similar as possible to those of humans. This wish, loosely expressing the desired behaviour of the robot, is translated into *design specifications*, which include the ability to coordinate dynamically and smoothly a primary and a secondary task. These design specifications are translated into a *design* (or *theoretical model*) that consists in a set of differential equations. Notably, this design is tested using a simulation environment developed by the authors of the study. The virtual environment integrates mechanisms for multi-robot control, multi-body dynamics, multi-contact multi-body resolution and haptic interaction for robot teleoperation (the integration modules presented in Sect. 2).

Kathib and colleagues describe a few simulation experiments carried out in the study. In one of them, the robot must maintain a fixed position with the left hand (primary task) while oscillating the left elbow (secondary task). Two different theoretical models are implemented and evaluated in simulation: a so-called dynamically consistent controller and a non-dynamically consistent controller. The two theoretical models produce different outputs in simulation. In particular, the dynamically consistent theoretical model is found to meet the design specification better than the non-dynamically consistent one. These outputs therefore enable the designers to formulate conditional predictions of the behaviour of the non-existing robot: if the robot implemented the dynamically consistent theoretical model, it would meet the design specification better than the non-dynamically consistent one. This evaluation—of the simulation outputs against the design specification—in turn, enables the authors to provisionally accept the hypothesis that the dynamically consistent theoretical model can produce the desired behaviour of the to-be-built robot. In this study, the simulation system is used for MO surrogative reasoning, i.e., to know how the system to be built would behave if it were governed by particular theoretical models, and is crucially involved in the discovery of the theoretical model that is sufficiently qualified to be moved forward to production.

The methodological reconstruction offered in this article has nothing to say on what comes after a theoretical model has been chosen, notably including the actual building of the theoretical model into a real-life system. However, the analysis of simulation models and surrogative reasoning made in the previous section enables one to reveal part of the methodological complexities involved in simulation-supported robotic design, which are somehow analogous to the complexities involved in the discovery of mechanisms via model-oriented surrogative reasoning in scientific research. First, the fact that a particular theoretical model is selected as a "good" model of the system to be built does not exclude that other theoretical models might have met the design specifications equally well. This is analogous to MO SsSR in science: this form of surrogative reasoning enables one to conclude, at most, that a particular theoretical

model can produce the behaviour of interest, but it does not provide conclusive reasons to exclude that other theoretical models might be as effective. This underdetermination problem is of some importance in scientific research (especially if the ultimate goal is to formulate a good mechanistic explanation of the behaviour of the target system), whilst it may be less serious in robotic design, since designers may well be content with arriving at *one* mechanism that is sufficiently effective in producing the desired behaviour. Second, whether a particular theoretical model is selected as a "good" model of the system to be built or not crucially depends on the content of the design specifications. Adding, removing, or changing the content of the design specifications may change "good" theoretical models into "bad" ones and vice versa. Similarly, in scientific research, whether a theoretical model can be regarded as a "good" model of the mechanism producing the behaviour of the target system crucially depends on how the latter is defined (e.g., on what factors concur with its specifications, or on its level of detail and approximation).

Third, and more crucially, the output of the simulation system is affected not only by the structure and content of the theoretical model, but also, as specified in Sect. 2, by several additional factors and modelling choices which are not constrained by the theory. The simulation model will typically implement an adjusted version of the theoretical model, a number of integration modules which are needed for the system to run, and a model of the environment. Therefore, if the simulation system meets (or fails to meet) the design specifications, one is not authorized to conclusively praise (or blame) the theoretical model: the reason for the success (or the failure) may lie in one or more of the many theoretically unconstrained aspects of the simulation system. Again, this issue is analogous to a known methodological problem affecting model-oriented computer simulations in scientific research, and implied for instance by the analysis by (Durán, 2020): a simulation system may (fail to) reproduce the behaviour of the target system for reasons that are unconnected to the structure and content of the theoretical model under scrutiny. This is a particularized version of the well-known Duhem-Quine problem affecting scientific experimentation at large.

## 4 Concluding remarks

Philosophers of engineering and technology have recently emphasized and discussed the fact that the design of (new) technological artefacts involves epistemic processes. Starting from this consideration, this article takes a first step in unravelling the structure of the epistemic processes involved in the design of new robotic systems, with a particular focus on the use of computer simulations. It has been argued here that, in this context, computer simulations are used to perform surrogative reasoning about the system to be built. More specifically, they are used to obtain conditional predictions about its behaviour—i.e., to predict the behaviour that the robot would generate if it was governed by a particular design. These conditional predictions are evaluated against the design specifications that are expected to be met by the final system, and the result of this evaluation is used to accept or reject a hypothesised design.

The structure of this epistemic process resembles to some extent the structure of model-oriented surrogative reasoning as it is frequently performed in scientific

research. MO SsSR plays an important role in the discovery of biological mechanisms, as discussed in (Datteri, 2020). Similarly, computer simulations can be reasonably regarded as tools for the discovery of "good" designs of robotic systems, a point that is not usually discussed in the current debate. This article thus forcefully affirms the presence and importance of epistemic processes in robotics in line with van Eck (2016). Moreover, it takes a further step revealing the structure of these epistemic processes in fields, such as robotics, in which this is far from evident. Computer simulations are increasingly used in all branches of scientific research, and there are good reasons to believe that methodologically sensible simulation studies can provide evidential grounds to theorise about the world (Humphreys, 2004; Weisberg, 2013; Winsberg, 2003). The same reasons may be invoked to believe that they can be epistemically useful to theorise the behaviour of a non-existing system and discover the mechanism that might comply with the design specifications. What fine-grained norms of rationality must guide methodologically sensible simulation-supported surrogative reasoning in robotic design is a question to be addressed in future studies by carrying out more detailed analyses of the inferential processes discussed here.

## Declarations

## References

Amigoni, F., & Schiaffonati, V. (2017). Models and experiments in robotics. In L. Magnani & T. Bertolotti (Eds.), *Springer handbook of model-based science* (pp. 799–815). Springer.

Boon, M., & Knuuttila, T. (2009). Models as epistemic tools in engineering sciences. In *Philosophy of technology and engineering sciences* (Issue March 2009, pp. 693–726). Elsevier. https://doi.org/10.1016/B978-0-444-51667-1.50030-6

Choi, H., et al. (2021). On the use of simulation in robotics: Opportunities, challenges, and suggestions for moving forward. *PNAS, 118*(1), 1–9. https://doi.org/10.1073/pnas.1907856118

Collins, J., Chand, S., Vanderkop, A., & Howard, D. (2021). A review of physics simulators for robotic applications. *IEEE Access, 9*, 51416–51431.

Contessa, G. (2007). Scientific representation, interpretation, and surrogative reasoning. *Philosophy of Science, 74*(1), 48–68. https://doi.org/10.1086/519478

Cummins, R. (1977). Programs in the explanation of behavior. *Philosophy of Science, 44*(2), 269–287. https://doi.org/10.1086/288742

Cummins, R. (1983). *The nature of psychological explanation*. The MIT Press.

Datteri, E. (2020). Biological accuracy in large-scale brain simulations. *History and Philosophy of the Life Sciences, 42*(1), 5. https://doi.org/10.1007/s40656-020-0299-1

Datteri, E., & Schiaffonati, V. (2019). Robotic simulations, simulations of robots. *Minds and Machines, 29*(1), 109–125. https://doi.org/10.1007/s11023-019-09490-x

Durán, J. M. (2020). What is a simulation model? *Minds and Machines, 30*(3), 301–323. https://doi.org/10.1007/s11023-020-09520-z

Franssen, M. (2020). Engineering design and the quest for optimality. In D. Michelfelder & N. Doorn (Eds.), *The Routledge handbook of the philosophy of engineering* (pp. 97–110). Routledge.

Frigg, R. (2010). Models and fiction. *Synthese, 172*(2), 251–268. https://doi.org/10.1007/s11229-009-9505-0

Frigg, R., & Nguyen, J. (2017). Models and representation. In L. Magnani & T. Bertolotti (Eds.), *Springer handbook of model-based science* (pp. 49–102). Springer. https://doi.org/10.1007/978-3-319-30526-4_3

Frigg, R., & Reiss, J. (2009). The philosophy of simulation: Hot new issues or same old stew? *Synthese, 169*(3), 593–613. https://doi.org/10.1007/s11229-008-9438-z

Galle, P. (1999). Design as intentional action: A conceptual analysis. *Design Studies, 20*(1), 57–81. https://doi.org/10.1016/S0142-694X(98)00021-0

Hartmann, S. (1996). The world as a process: Simulations in the natural and social sciences. In R. Hegselmann, U. Mueller, & K. G. Troitzsch (Eds.), *Modelling and simulation in the social sciences from the philosophy of science point of view, theory and decision library* (pp. 77–100). Kluwer.

Hay, E., Hill, S., Schürmann, F., Markram, H., & Segev, I. (2011). Models of neocortical layer 5b pyramidal cells capturing a wide range of dendritic and perisomatic active properties. *PLoS Computational Biology*. https://doi.org/10.1371/journal.pcbi.1002107

Humphreys, P. (1990). Computer simulations. *PSA: Proceedings of the Biennial Meeting of the Philosophy of Science Association, 2*, 497–506.

Imbert, C. (2017). Computer simulations and computational models in science. In L. Magnani & T. Bertolotti (Eds.), *Springer handbook of model-based science* (pp. 735–781). Springer. https://doi.org/10.1007/978-3-319-30526-4_3

Khatib, O., Sentis, L., Park, J., & Warren, J. (2004). Whole-body dynamic behavior and control of human-like robots. *International Journal of Humanoid Robotics, 1*(1), 29–43. https://doi.org/10.1142/S0219843604000058

Kroes, P. (2021). Engineering design. In D. Michelfelder & N. Doorn (Eds.), *The Routledge handbook of the philosophy of engineering* (pp. 289–299). Routledge.

Johnson-Laird, P. N. (1983). *Mental models*. Harvard University Press.

Liu, C. K., & Negrut, D. (2021). The role of physics-based simulators in robotics. *Annual Review of Control, Robotics, and Autonomous Systems, 4*, 3:1-3:4. https://doi.org/10.1146/annurev-control-072220-093055

Massimi, M. (2019). Two kinds of exploratory models. *Philosophy of Science, 86*(5), 869–881. https://doi.org/10.1086/705494

Michelfelder, D., & Doorn, N. (Eds.). (2021). *The Routledge handbook of the philosophy of engineering*. Routledge.

Ord-Smith, R., & Stephenson, J. (1975). *Computer simulation of continuous systems*. Cambridge University Press.

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

van de Poel, I. R., & Royakkers, L. M. M. (2011). *Ethics, technology, and engineering: an introduction*. Wiley-Blackwell.

Poznic, M. (2016). Modeling organs with organs on chips: Scientific representation and engineering design as modeling relations. *Philosophy & Technology, 29*(4), 357–371. https://doi.org/10.1007/s13347-016-0225-3

Reimann, M., Anastassiou, C., Perin, R., Hill, S. L., Markram, H., & Koch, C. (2013). A biophysically detailed model of neocortical local field potentials predicts the critical role of active membrane currents. *Neuron, 79*(2), 375–390. https://doi.org/10.1016/j.neuron.2013.05.023

Rescorla, M. (2014). A theory of computational implementation. *Synthese, 191*(6), 1277–1307. https://doi.org/10.1007/s11229-013-0324-y

Simon, H. A., & Newell, A. (1962). Computer simulation of human thinking and problem solving. *Monographs of the Society for Research in Child Development, 27*(2), 137. https://doi.org/10.2307/1165535

Swoyer, C. (1991). Structural representation and surrogative reasoning. *Synthese, 87*(3), 449–508. https://doi.org/10.1007/BF00499820

van Eck, D. (2015). Mechanistic explanation in engineering science. *European Journal for Philosophy of Science, 5*(3), 349–375. https://doi.org/10.1007/s13194-015-0111-3

van Eck, D. (2016). *The philosophy of science and engineering design*. Springer.

Vermaas, P., Kroes, P., van de Poel, I., Franssen, M., & Houkes, W. (2011). A philosophy of technology: from technical artefacts to sociotechnical systems. *Morgan and Claypool Publishers*. https://doi.org/10.2200/S00321ED1V01Y201012ETS014

Weisberg, M. (2013). *Simulation and similarity: Using models to understand the world*. Oxford University Press.

Winsberg, E. (2003). Simulated experiments: Methodology for a virtual world. *Philosophy of Science, 70*(1), 105–125. https://doi.org/10.1086/367872

Winsberg, E. (2009). Computer simulation and the philosophy of science. *Philosophy Compass, 4*(5), 835–845. https://doi.org/10.1111/j.1747-9991.2009.00236.x

Yaghmaie, A. (2021). Scientific modeling versus engineering modeling: Similarities and dissimilarities. *Journal for General Philosophy of Science, 52*(3), 455–474. https://doi.org/10.1007/s10838-020-09541-3

Žlajpah, L. (2008). Simulation in robotics. *Mathematics and Computers in Simulation, 79*(4), 879–897. https://doi.org/10.1016/j.matcom.2008.02.017