

Exploring a Mechanistic Approach to Experimentation in Computing

Eric Hatleback · Jonathan M. Spring

Received: 3 December 2013 / Accepted: 10 April 2014 / Published online: 25 July 2014
© The Author(s) 2014. This article is published with open access at Springerlink.com

Abstract The mechanistic approach in philosophy of science contributes to our understanding of experimental design. Applying the mechanistic approach to experimentation in computing is beneficial for two reasons. It connects the methodology of experimentation in computing with the methodology of experimentation in established sciences, thereby strengthening the scientific reputability of computing and the quality of experimental design therein. Furthermore, it pinpoints the idiosyncrasies of experimentation in computing: computing deals closely with both natural and engineered mechanisms. Better understanding of the idiosyncrasies, which manifest in terms of a nonstandard role for experimentation, are interesting both for computer scientists and for philosophers of science. Computer scientists can think more clearly about their experimental choices. The role of experimentation elucidated by computer science merits further study from philosophers of science generally, as it highlights a role for experimentation hitherto unrecognized by philosophers: demonstration that *activities* exist.

Keywords Computing · Experiments · Mechanisms · Experimental method

1 Introduction

This note is largely a rejoinder to a particular portion of the work by Viola Schiaffonati and Mario Verdicchio recently published in this journal (Schiaffonati and Verdicchio 2013). They address the status of experimentation in computing, concluding (in part) that “calling for experiments in computing does not equate with making computing

E. Hatleback (✉)

Department of History and Philosophy of Science, University of Pittsburgh, 1017 Cathedral of Learning,
4200 Fifth Avenue, Pittsburgh, PA 15260, USA
e-mail: enh8@pitt.edu

E. Hatleback
e-mail: hatleback@gmail.com

E. Hatleback · J. M. Spring
CERT® Coordination Center, Software Engineering Institute, Carnegie Mellon University,
4500 Fifth Avenue, Pittsburgh, PA 15213-2612, USA

more similar, at least from a methodological point of view, to traditional scientific disciplines” (Schiaffonati and Verdicchio 2013, p. 15). Here, we dispute that claim: experimentation in computing can benefit from making strides toward similarity with traditional scientific disciplines. Such an attitude—when warranted—reveals both a positive direction for the status of experimentation in computing and a new role for experimentation for philosophers of science to consider.

Building upon the work done by Schiaffonati and Verdicchio, we argue that experimenters in computing should adopt the mechanistic approach that has gained favor in the philosophy of science literature.¹ Doing so, we claim, reveals that experimentation in computing breaks into two categories. The first, which deals with the natural operations of systems, is a straightforwardly physical endeavor.² The mechanisms under investigation in experiments of this type are labeled *physical mechanisms*. Such experiments are likened to mechanistic experimentation in biology. Experimentation of the second type, which deals with the generation of artifacts, the interaction between systems, or human exploitation of an artifact or system, is intimately tied to engineering endeavors. Thus, the second category deals with the engineered operations of systems. The mechanisms under investigation in experiments of this type are labeled *engineered mechanisms*.

In Section 2, the progress made by Schiaffonati and Verdicchio is presented. The aim is elucidation of common ground for further analysis in the subsequent sections. In Section 3, it is argued that experimentation on physical mechanisms in computing bears resemblance to mechanistic experimentation in biology.³ Thus, in contrast to the urging of Schiaffonati and Verdicchio, experimentation of this type in computing does warrant comparison to (and should strive for similarity with) the methodology of a traditional scientific discipline. Section 4 introduces Allan Franklin’s classification of the roles of experimentation in science. Franklin’s work is included to substantiate the claim that experimentation on engineered mechanisms in computing introduces a role for experimentation that is worthy of extended analysis by philosophers of science. Concluding remarks are contained in Section 5.

2 Bisecting the Role of Experimentation in Computing

Schiaffonati and Verdicchio contribute to our understanding of the role of experimentation in computing.⁴ In particular, their work highlights an important feature of computing: it has an intrinsic engineering component. Ignoring the engineering

¹ In Sect. 2, we will provide the necessary details and definitions from the mechanistic approach. Furnishing a complete account of the emergence of the contemporary mechanistic approach in the philosophy of science literature is beyond the scope of this paper. One good starting point for undertaking that endeavor is the entry on mechanisms (and the references contained therein) in *The Routledge Companion to Philosophy of Science* (Glennan 2014).

² Examples might include the electrical conductivity of a material or the way that the human eye perceives changes to shapes on a screen.

³ In maintaining the mechanistic approach, an underlying assumption is that human psychological factors are rooted in physical mechanisms. See the keystroke study reported in Killourhy and Maxion (2009) for an example of the interplay between computing and psychological states as physical mechanisms.

⁴ “Computing” is used broadly; as Schiaffonati and Verdicchio note, the term is nebulous. We will implement it with the same intended denotation as it is used in their article. See pp. 2–4 of their work for details.

component can, in some cases, generate discord when one attempts to liken experimentation in computing to experimentation in the established sciences:

The proposals that are endowed with a discussion on concrete examples show that the application of general experimental principles is not straightforward and immediate, and reveal that differences with traditional sciences are not due only to the relatively young age of computing, but also to its intrinsically different disciplinary nature, scientific and, at the same time, engineering. (Schiaffonati and Verdicchio 2013, p. 14)

According to this account, properly noting the engineering component of computing amounts to being aware that experiments involving it have “the goal of demonstrating that a given artifact is working in some way or that it is better than another. [...] Experiments about artifacts in computing tell us more about the people that have done the job, than the way the world is” (Schiaffonati and Verdicchio 2013, pp. 14–15). Schiaffonati and Verdicchio argue against proposals that aim to overhaul experimental practice in computing through conforming to experimental practice in the established sciences. They reason that properly accounting for the engineering influence in computing makes conformity with the experimental practices of the established sciences misguided.

The distinction is important, but it stands in need of refinement. Clearly Schiaffonati and Verdicchio do not intend to imply that engineering cannot learn from science, nor that science has no place informing engineering decisions. However, the claim that an engineering discipline should eschew the experimental methods of traditional science appears to imply exactly that. Schiaffonati and Verdicchio themselves recognize that further development is needed, as they close their paper by noting that the results “call for a rethinking” and that they have “singled out some open issues” (Schiaffonati and Verdicchio 2013, pp. 15–16). The following is meant to initiate the refining process.

Philosophers of science have contributed substantially to our understanding of scientific methodology. Of particular importance for the present work is the mechanistic account. Peter Machamer, Lindley Darden, and Carl Craver co-authored a seminal paper on mechanisms in 2000.⁵ In it, the components of a mechanism are identified as entities and activities, and mechanisms themselves are defined accordingly: “Mechanisms are entities and activities organized such that they are productive of regular changes from start or set-up to finish or termination conditions” (Machamer et al. 2000, p. 3). For this project, we will adopt that definition. This ontological framework offers the potential for conceptual clarity regarding the role of experimentation in computing: successful experimentation on this account would amount to progress toward understanding how the mechanism in question works.⁶

⁵ As Glennan notes, “Machamer et al. (2000) is the most widely cited paper on mechanisms” (Glennan 2014, p. 428). Accordingly, in what follows, we will rely predominantly on that article as representative of the mechanistic approach.

⁶ For a related discussion involving a mechanistic account of computation, see especially Piccinini (2007). Other discussions involving accounts of computation include Piccinini and Scarantino (2011), Fresco (2013), Scheutz (1999), Copeland (1996), and Fresco and Wolf (2013).

The regular operation of a mechanism is one of its indispensable features:

Mechanisms are regular in that they work always or for the most part in the same way under the same conditions. The regularity is exhibited in the typical way that the mechanism runs from beginning to end; what makes it regular is the *productive continuity* between stages. Complete descriptions of mechanisms exhibit productive continuity without gaps from the set up to termination conditions. Productive continuities are what make the connections between stages intelligible. If a mechanism is represented schematically by $A \rightarrow B \rightarrow C$, then the continuity lies in the arrows and their explication is in terms of the activities that the arrows represent. A missing arrow, namely, the inability to specify an activity, leaves an explanatory gap in the productive continuity of the mechanism. (Machamer et al. 2000, p. 3, emphasis in original)

Exploiting this attitude with respect to experimentation in computing has potentially large payoffs for the quality of work and general scientific respectability of the field.⁷ Systems in computing are mechanisms, so an experimental approach designed to explicate the system (mechanism) at hand results in clarity, reproducibility, and transferability of results.⁸

Whereas Schiaffonati and Verdicchio advocate a conceptual dualism in computing comprised of its “scientific” and “engineering” components, a more fruitful approach is to split experimentation in computing into *physical* and *engineered* categories, according to whether the mechanism of the system under exploration is natural (physical) or artificial (engineered). The switch is not merely semantic. Instead, it removes the insinuation that the “engineering” and “scientific” features of computing are distinct from each other. Rather, computing *as a discipline* exhibits scientific potential, and that potential can approach realization through experimentation dealing with either a computing system’s physical facet or its engineered facet.⁹

For an intuitive heuristic to make the distinction, one may use the “speed” at which the mechanism in question changes. Physical mechanisms change at geological (or perhaps even cosmological) speed: the entities and activities operate identically for long stretches of time. Long-term changes may occur in the mechanism’s operation, but those changes take significant time. For example, the reproductive mechanism of a certain species may gradually evolve through natural selection, but for generations-long stretches of time it remains the same. And the fundamental physical forces (such as the electromagnetic force) may never change, or may change so slowly that their values can be assumed fixed. Natural physical mechanisms always operate the same during the time involved in any computing research project.

⁷ By “scientific respectability,” we refer loosely to the longstanding debate discussed at length by Schiaffonati and Verdicchio in the opening to their paper. For evidence that the scientific respectability of computing is in need of improvement, see (for example) the references provided in footnote 31 below.

⁸ Further argumentation for the claim that systems in computing are mechanisms is presented in Sect. 3.

⁹ A discourse on precisely what qualities are necessary for a discipline to exhibit in order for it to count as scientific would take us too far afield. We will assume that experimentation is a necessary feature of a scientific discipline. Accordingly, to the extent that the quality of experimentation in computing is subpar, so too will its scientific status remain subpar. Additionally, an interesting case that warrants study beyond the present work is emergent behavior. It is not obvious whether either class (physical or engineered) fits scientific experimentation on emergent behavior.

Engineered mechanisms, on the other hand, change at “human” speed: the operation of the entities and activities might change very quickly. Internet governance standards may be developed and deployed, for example, that directly and immediately impact the operation of a computing mechanism with respect to security or privacy. And the mechanism involved in a network connection changes rapidly as the users of the connection impart changes to it. Engineered mechanisms are vulnerable to rapid changes in the way that their entities and activities operate, whereas physical mechanisms are not vulnerable to the same sort of change.

This is not to say that a system involving a physical mechanism does not exhibit change. Activities, a key element of a mechanism, are “the producers of change” (Machamer et al. 2000, p. 3). The growth or disease of an organism is a change, and each type of disease, for example, has its own mechanism, but the mechanism by which any particular disease type makes any particular type of organism ill does not change over the course of investigation. Bacteria and viruses are also known to evolve quickly—perhaps even at “human” speed—but the mechanism of evolution does not change, nor does the mechanism of protein transcription or the action of the new proteins within the cell. New entities, perhaps in the form of genetic material and proteins, are introduced, but the mechanism remains coherent, and the activities largely remain the same in physical systems.

A more refined definition of engineered mechanisms captures both the intuition of quick mechanism change and the possible conflation with speedy elements of the physical world: engineered mechanisms are susceptible to having their entities or their activities changed during the course of the investigation at the will of a rational decision-making entity. Physical mechanisms do not have this property.

Such a distinction makes possible a consistent viewpoint on experimentation in computing that accounts for the claims of Schiaffonati and Verdicchio. Whereas they characterize the “engineering” component as inherently present in all computing experiments, this need not be the case.¹⁰ Many experiments in computing isolate only a physical mechanism within a system.¹¹ Though many artifacts are designed to take advantage of certain physical mechanisms, care is necessary for proper classification. For example, merely the fact that humans can build a spring does not mean that the spring constant or the model for forces exerted by a spring have been built as well; those are physical mechanisms (or representations of physical mechanisms) to which the artifact is subject. The same holds true for computers. Transmitters or human-facing displays (for example) may be artifacts that take advantage of, or are optimized for, a particular physical mechanism. Experimentation likely is helpful in determining which

¹⁰ See Schiaffonati and Verdicchio, p. 14, for their insinuation that the engineering component of computing resonates throughout experimentation in computing: “We claim that the engineering component of computing should be considered when reflecting on the nature and role of experiments in computing.” This suggests that the “engineering” component is omnipresent in computing, and therefore that it warrants consideration in all computing experiments.

¹¹ One example is determining a transmission channel’s signal-to-noise ratio. Shannon’s work on this problem is fundamental to computing. However, the SNR bounds are independent of any device. Another example is determining the psychological parameters of human–computer interaction (HCI) such as fighter pilot situational awareness based on varying the computer’s heads-up display (HUD) design. In this case, the mechanism in question really relates to how the human brain processes information, which is not an engineered mechanism.

mechanisms are relevant and in measuring their effects. Clearly distinguishing these possible confounding situations should improve experimental design in computing.

Furthermore, the outlined distinction nicely situates computing within the landscape of science. Every scientific discipline shares with the others a grounding in physical phenomena. Even still, lest they be the same discipline, a necessary distinction likewise exists between any pair of sciences. For computing, the grounding in physical phenomena comes through experimentation that demonstrates the physical operation of systems and the psychological aspects of human computer users. However, its distinction from other sciences emerges from its *engineered* facet. It is in this context that the argumentation provided by Schiaffonati and Verdicchio rings true. However, they have overstated the distinction by implying that this fact means that computing could not improve its experimental methodology by learning from other sciences.¹²

Its distinction—the engineered facet—provides computing with its own scientific territory. For example, life (broadly construed) stands as the distinguishing feature of biology, and consciousness (broadly construed) stands as the distinguishing feature of psychology.¹³ Each of those scientific disciplines admits of experimentation involving the physical phenomena under its purview, but it also admits of experimentation involving its “distinguishing feature”—essentially, the reason why it is useful to call the field of study by its own name.

Computing further subdivides into myriad specialized factions, just as biology and psychology each admit of many specialized subfields of study. Cybersecurity, for example, focuses on the interplay of criminology, cryptography, economics, and computing. Experimentation in cybersecurity thus can and should emphasize one of two main subjects: physical phenomena and engineered phenomena. Physical phenomena involved in cybersecurity research include, for example, electromagnetic interference with communication channels (active jamming (Shenghua et al. 2004), passive jamming (Reade et al. 2007), or EMP disruption (St. Sauver 2009)) or the consistency of human keystroke typing patterns used for biometric identification (Killourhy and Maxion 2009). Engineered phenomena involved in cybersecurity research include, for example, socially interactive deception based on website design choices (Dhamija et al. 2006) or DNS cache poisoning attacks due to UDP port number or Query ID selection algorithms (Friedl 2008).

In both physical and engineered cases, these mechanistic descriptions would involve products that are the result of engineering—keyboards, for example. However, the question of whether or not humans type differently from each other, given a keyboard, is not subject to mechanistic change during the course of investigation. A particular cache poisoning attack script also uses products of engineering, but the mechanism under study is itself *engineered*, and it is changeable. The crucial distinction is not whether products of human engineering are utilized in the inquiry, but whether the mechanism under investigation has been engineered. For a mechanism to be engineered, the entities and the activities involved in the mechanism must be engineered.

¹² See once more the quote from Schiaffonati and Verdicchio, p. 15, for evidence: “[W]e hope to have made clear that calling for experiments in computing does not equate with making computing more similar, at least from a methodological point of view, to traditional scientific disciplines.”

¹³ By “distinguishing feature,” we mean that particular focus that separates biology (for example) from chemistry (again, for example).

The following extended example is intended to illustrate the qualities of engineered mechanisms. An experimenter is presented with an odd-looking calculator with only 5 buttons: “1,” “2,” “+,” “-,” and “ENTER”. Reminiscent of an oddity one might find in one of Andy Warhol’s time capsules, let us name the device “Warhol’s Calculator.” Warhol’s Calculator is designed so that each button, when pressed, relays the contents of a digital storage element (e.g., a file or register) to a cache, and when ENTER is pressed the calculator displays the results. The calculation mechanism embodied by the calculator works with entities {1, 2} and activities {addition, subtraction}. While Warhol’s Calculator, the device, appears to be the product of human engineering, this does not directly bear on whether we call the mechanism *engineered*.

To determine whether the mechanism is engineered, we must determine whether the entities and activities of the mechanism can be changed by rational agents. Imagine that the user of Warhol’s Calculator presses 1+1ENTER and the display reads “2,” as expected. The user repeats one second later, and the display reads “4.” The user repeats again one second later, and the display reads “0.” What the user does not know is that the calculator’s entities and activities are being changed, altering the mechanism that executes before the display. When the user presses 1, Warhol’s Calculator sends the contents of the file associated with this button. But Warhol’s Calculator is designed to swap the contents of the files associated with the buttons “1” and “2” every second. The activities also change: the algorithm associated with “+” is swapped with that associated with “-” every two seconds. So, in the first iteration, Warhol’s Calculator calculates 1+1, but then 2+2, and, in the third iteration, 1-1.

In the natural sciences, where physical mechanisms are under investigation, the entities and activities cannot be so manipulated. For example, using artificial selection, humans can breed very odd-looking birds. Though we may alter certain entities (perhaps which pigeons are allowed to breed), the entities and activities of the underlying, finer-grained mechanism of chromosome replication during meiosis cannot be altered by the rational agent.

Warhol’s Calculator, an engineered mechanism, is different. The entities and activities are changeable by a rational agent. The original instantiation involved each entity alternating every second and each activity alternating every 2 s, but this itself was a design choice by a rational agent. Warhol himself may be sitting behind the experimenter with a toggle switch, alternating the entities and activities of the mechanism at his discretion. This makes the experimenter’s job practically much more difficult: she must design experiments to account for changeable, *engineered* mechanisms. This is the essential importance of designating mechanisms as physical or engineered: engineered mechanisms are different from physical mechanisms owing to their susceptibility to be changed by a rational agent. Accordingly, experimental design for the investigation of engineered mechanisms may be fundamentally different from experimental design for the investigation of physical mechanisms.

Given the distinctness of physical and engineered mechanisms and the development of contemporary science into extremely fine-grained areas of specialization, any given scientific experiment performed in computing should be classifiable exclusively as a physical computing experiment *or* as an engineered computing experiment. An experiment that dabbles in both areas almost certainly is too coarse-grained for its results to be qualified as scientific. Such an experiment, by virtue of testing both physical and

engineered facets, would necessarily involve a degree of freedom in both domains, thereby casting doubt on the scientific integrity (specifically, the internal validity) of its results.

The urging by Schiaffonati and Verdicchio to view computing as inherently involving engineering is fruitful. However, deeper clarity is afforded by noting, first, the foregoing distinction between *engineering* and *engineered*, and second, that any specific scientific experiment in computing should target *either* a physical mechanism within a system *or* an engineered mechanism within a system.¹⁴ The following two sections aim to identify some of the implications of viewing experimentation in computing in these terms. A mechanistic approach to experimentation in computing reveals that computing can benefit from emulating the methodology of an established science, especially when the experimentation targets a physical mechanism within a system.¹⁵ Furthermore, philosophers of science can benefit from studying experimentation in computing, particularly experimentation on engineered mechanisms.¹⁶

3 Experimentation on Physical Mechanisms in Computing

Experiments in computing that isolate physical mechanisms stand to benefit from adopting the outlook of traditional sciences toward experimentation. This claim will be supported first by likening physical systems in computing to mechanisms in experimental biology. Then, after extracting a set of desiderata from experimental biology, we see how adopting those desiderata would benefit at least one subdiscipline of computing, namely cybersecurity. This approach, though subject to the known limitations of case studies, will nonetheless provide initial support for the claim that computing experiments that isolate a physical mechanism could benefit from adopting the methodology employed in established sciences.

3.1 Case Study: Biology

The mechanistic ontology, composed of entities and activities, has contributed clarity to our understanding of physical experimentation. For example, the flexibility for mechanisms to be descriptively coarse-grained or fine-grained is a virtue. Machamer, Darden, and Craver write:

Scientists do not always provide complete descriptions of mechanisms at all levels in a nested hierarchy. Also, they are typically interested in types of

¹⁴ Importantly, this claim is meant to apply to good experiments in computing. As will be seen, the literature is rife with examples of bad experimentation, and often what makes those experiments bad is a failure to identify a single experimental mechanism.

¹⁵ It is important to note that computing experiments that target engineered mechanisms often stand to gain from employing the methodology of established sciences as well. However, the benefit is most clearly exemplified for physical mechanisms, as will be seen.

¹⁶ Identifying the distinction between physical mechanisms and engineered mechanisms in computing is a task separate from determining the individual roles played by physical and engineered mechanisms in computing (and perhaps their interaction with each other in a joint role). Our task in Sect. 2 has been the former; we view that step as one that is necessary to complete before the latter can be attempted. A separate piece of substantial research will be necessary before progress on the latter step can be made.

mechanisms, not all the details needed to describe a specific instance of a mechanism...[Schema terms] may be characterized as activities in higher-level mechanisms. (Machamer et al. 2000, pp. 15–16)¹⁷

The “nesting” feature of mechanisms is important. It unites experimentation at different levels of granularity. Since the mechanistic approach takes phenomena to be exhibitions of the operations of mechanisms, experimental inquiry amounts to investigation of the mechanism(s) that produce(s) the phenomenon in question. The nesting feature of mechanisms thereby solves a problem described by Radder:

Since scientific practice does not consist of isolated experiments performed by solitary experimenters, we have to examine the ways in which individual experiments are embedded and used in broader experimental and theoretical contexts. (Radder 2003, p. 153).

The mechanistic approach accommodates the embedding. A coarse-grained mechanism is comprised of finer-grained mechanisms. Experimentation can occur at any level of granularity by designing the experiment to isolate the entity or activity in question at the grain size appropriate to the scope of inquiry.

In the computing context, this is particularly germane because systems in computing admit of precisely this feature of varying granularity. A network of computers is a system, but so is one portion of an individual machine. The granularity of the respective entities and activities under investigation in an experiment will vary according to the scope of the experimenter’s inquiry. Using this approach, the experimenter tailors the experimental design to the mechanism in question.

We see the very same approach to mechanisms in biology: the nesting of mechanisms in biology is particularly akin to the nesting of systems in computing. For example, an ecosystem is comprised of individual organisms, just as a network is comprised of individual machines. Likewise, an individual organism is comprised of unique organs which work together to permit the organism to function, just as an individual machine is comprised of unique components which work together to permit the machine to function. Experimentation in biology may isolate a mechanism at the ecosystem level, the organism level, the organ level, or still other levels of granularity. Similarly, experimentation in computing may isolate a system at the network level, the machine level, the component level, or still other levels of granularity.

Furthermore, biological mechanisms operate regularly owing to their physical grounding. This permits prediction. The desire for predictive power has not gone unnoticed in the cybersecurity domain: Maxion, Longstaff, and McHugh claim that “prediction is the key result that we want from scientific research” (Maxion et al. 2010, p. 2). When restricting our scope to physical computing mechanisms, we see that such prediction is attainable owing to the assurance that the mechanisms in question operate regularly over large stretches of time.¹⁸ In both biology and computing, the regular

¹⁷ The addition for clarity is bracketed.

¹⁸ A possible exception occurs in the case of networks, which operate based upon engineered control components (the code). Experimentation that isolates that code might not involve a physical mechanism of the sort under consideration in this section.

operation of mechanisms offers the capability for prediction, a quality valued in both domains.

Given the similarities between physical computing mechanisms and biological mechanisms, it is reasonable to expect that we could obtain guidance for reputable scientific experimentation on physical mechanisms in computing through studying the methodology employed in an established science such as experimental biology. To that end, we next seek a set of qualities that experimenters in computing could use to strengthen the foundation upon which their experimental endeavors are built.¹⁹

To begin the task of extracting desiderata that are already at work in reputable experimental research in biology, consider an arbitrary research experiment chosen from a recent issue of *The Journal of Experimental Biology*.²⁰ In the experiment, researchers sought to discover the mechanism for seafaring birds' ability "to pinpoint their breeding colony even within an apparently featureless environment" (Gagliardo et al. 2013). That the experimenters take the mechanistic approach outlined in Sect. 2 is apparent immediately. The second sentence of the article's abstract reads: "The mechanisms underlying their surprising navigational performance are still unknown" (Gagliardo et al. 2013, p. 2798). The experiment is thereby designed to isolate the physical mechanism that results in successful navigation for the birds.

The article is broken into three sections ("Materials and Methods," "Results," and "Discussion").²¹ Within the "Materials and Methods" section, the authors note that over the course of two separate trial runs of 24 birds each, 8 birds were equipped to determine whether magnetic-sensing entities in the brain were the primary operators in the mechanism, 8 birds were equipped to determine whether olfactory entities in the nose were the primary operators, and 8 birds were untreated as a control measure. These choices were made in accordance with previous research that demonstrated that the leading candidates for the mechanism are magnetic navigation and olfactory navigation.²² These settings for the experiment speak directly to the experimenters' desire to control independently for the two potentially existing variables in the mechanism under scrutiny. (In this case, the variables are entities, but they may be activities in other experimental structures.) Thus, we see that one desideratum is internal validity:

¹⁹ An alternative approach to the one being taken here might involve straightforwardly listing and adopting the principles already accepted as standards in experimental biology. A good introduction to those principles is provided in Weber (2012), especially Sect. 3. However, our intention is to provide a starting point from which computing might refine and develop its own set of principles. Accordingly, we believe that the better approach (as conveyed in the following) is to identify a core set of properties from a traditional science (in this case, biology), from which computing-specific principles can be developed. In this way, computing would benefit from the methodology of the established sciences without being wholly restricted to properties and principles that have already been tailored to the established sciences in question.

²⁰ The arbitrariness of the choice is deliberate, as will be addressed. The strategy is to provide the reader with a specific example for clarity while, at the same time, noting some general features of scientific experimentation in the field of biology.

²¹ There is an opening "Introduction" section that chiefly introduces the subject material and lightly describes the experiment, but the details are included in the subsequent sections, making the "Introduction" dispensable for the current analysis.

²² The references for the previous research leading to these two candidates were provided in the "Introduction" section by the authors.

Internal validity: the mechanism under experimentation is of suitable scope to achieve the reported results; the entities and activities of the experimental mechanism are not susceptible to systematic error.

An experiment can lack internal validity in two ways. First, it may be the case that too many entities or activities are unknown. Second, the granularity of the experimental mechanism might not match the granularity of the desired result.²³ In either situation, the experiment's result would not be sufficient to clarify the mechanism under scrutiny.

Also within the "Materials and Methods" section, the authors note the precise details of the birds implemented in the experiment:

During the 5 days prior to the experiment the colony was monitored daily, to check whether the incubating individuals had been relieved by their partner during the previous night. Only those individuals that had returned to the colony during the last two nights before the experiment were captured. This was done in order to displace only birds that were highly motivated to return to the nest, and with presumably no or low motivation to forage: indeed, after returning to the nest to relieve its partner during incubation, a bird normally fasts during the 7–10 days it spends on its egg. (Gagliardo et al. 2013, p. 2799)

This procedure speaks to a second desideratum. Not only were actual members of the "wild" subjects used; they were chosen with forethought that ensured that the experimental mechanism would be faithful to the nonexperimental mechanism. Had they chosen birds that potentially needed to forage, it would be unclear whether the navigational mechanism observed in the experiment would represent the navigational mechanism "in the wild." This speaks to the experiment's external validity:

External validity: the mechanism under experimentation (and therefore the result of the experiment) is not solely an artifact of the laboratory setting; the experimental mechanism is faithful to the mechanisms "in the wild."

The careful selection of subjects outlined in the passage above also contributes to a third desideratum. The need for food could have been a pre-mechanism cause that confounded the experimental results, since birds foraging for food are likely to use the navigation mechanism (or perhaps an entirely different mechanism) to hunt for food rather than use the targeted navigational mechanism to return to the colony. Additionally, careful notes throughout the "Materials and Methods" section detail the postexperimental consequences as safe for the subjects. Examples include the facts that the eggs of the captured birds that needed incubating were replaced with plaster eggs to prevent confusion in the colony, that the real eggs were incubated properly until the conclusion of the experiment, and that the nasal wash used to eliminate the olfactory cues of the birds had an effective duration of only a few weeks before complete functionality returned (Gagliardo et al. 2013, p. 2799). These factors all speak to the containment of the experiment:

²³ An example of this sort of "granularity mismatch" might involve trying to acquire experimental results about interaction at the cellular level with an experimental mechanism at the ecosystem level.

Containment: no pre-mechanism causes threaten to confound the results, and no postmechanism effects are a threat to safety.

Finally, within the “Results” section, the reader can find a complete account of the experiment’s outcome and how those results contribute to understanding the mechanism under investigation. In this case, the provided results yield strong support for the hypothesis that the navigational mechanism includes olfactory entities as necessary pieces of the mechanism. It is here, and within the ensuing “Discussion” section, that the experiment’s contribution toward understanding the mechanism is detailed. Explicit discussion of the results contributes transparency to the experiment, since such discussion contributes to the completeness of the diagram for the mechanism under study. Optimal transparency for the mechanism under investigation is sought throughout the report. Accordingly, a fourth desideratum is transparency:

Transparency: there are no explanatory gaps in the experimental mechanism; the diagram for the experimental mechanism is complete.

Transparency involves clarity in describing the experimental mechanism’s parameters. Without that clarity, the experiment will not be capable of replication or reproduction.

Lest the reader think that the specific article under consideration is unique, the research articles in *The Journal of Experimental Biology* each contain precisely the same named sections: “Introduction,” “Results,” “Materials and Methods,” and “Discussion.”²⁴ This serves to maintain a standard level of scientific reputability, both for the journal and for the discipline. Likewise, the structure serves to avoid violation of the desiderata (internal validity, external validity, containment, and transparency) identified in the case study presented here.²⁵

3.2 Case Study: Cybersecurity

In contrast to the standardized approach in experimental biology, the experimental literature in cybersecurity stands in need of scientific direction. There is no standard format for reporting experimental results.²⁶ Indeed, upon selecting what appears to be the most replicable experiment from among the publications in the 2013 IEEE

²⁴ This can be verified by inspecting the articles published in the journal. At the time of this publication, the journal’s recent (and current) research articles all exhibit these distinct sections. Furthermore, the submission policies for 2014 note a switch in the presentation order of the “Materials and Methods” section and the “Discussion” section: the “Discussion” section now appears before the “Materials and Methods” section. This is strong evidence for the necessary inclusion of these sections in all research article submissions.

²⁵ Surely other experimental virtues are striven for in experimental biology; we do not intend for the set to be interpreted as exhaustive. Instead, the intent is to identify a strong, uncontroversial set of desiderata that cybersecurity (as a case study from computing) could benefit from adopting. As will be described in Sect. 3.2, if this core set were widely adopted in cybersecurity (and suitably adjusted), it would contribute a great deal to the cybersecurity literature on experimentation.

²⁶ See, for example, the full set of proceedings from the 2013 IEEE Symposium on Security and Privacy, which is one of the most prestigious conferences (and publications) in the field of cybersecurity. The articles, even though they appear within the same publication, admit of no similarities in structure to each other save for the inclusion of an “Introduction” section within each contribution.

Symposium on Security and Privacy, attempts to complete the replication have been met with great difficulty.²⁷

The foregoing suggests that a legitimate starting point might involve reformulating the four desiderata in place for biological experimentation for use in the cybersecurity domain. For example, adding security concerns to the Containment desideratum would tailor its application to cybersecurity while retaining the methodological clout that the desideratum delivers to methodology in experimental biology. Other minor adjustments could be imparted to the desiderata as needed to tailor them to the various subdisciplines of computing. Doing so would contribute to the scientific respectability of cybersecurity and the other subfields of computing.

Some recognize the state of disrepute of published experimental work in the cybersecurity field. For example, Roy Maxion and his colleagues have several contributions in the literature that urge for the adoption of standard experimental practices from traditional sciences.²⁸ The experimentation with biometric security measures performed by Kevin Killourhy and Maxion is particularly noteworthy (Killourhy and Maxion 2009). In that study, the physical mechanism involved with keystroke dynamics receives experimental attention. Furthermore, the study meets all four of the outlined desiderata: equipment capable of capturing the needed measurements and a series of algorithms were used to assess the data (internal validity); the subjects were varied in age, gender, and handedness (external validity); the apparatus measured only the keystrokes entered during the trials and ceased after a successful run, and the experimental design did not harm the human subjects (containment); and Killourhy and Maxion detail each entity and activity involved in the experimental setup (transparency).²⁹ They also publish further details and the complete set of collected data on a website so that it may be analyzed by any who wish to do so.

Like Killourhy and Maxion, Christian Rossow and colleagues have argued for an overhaul in experimental practice, but in the malware research subfield.³⁰ Rossow and his colleagues identify a general failure of sound experimental method—what they label a lack of “prudence”—in the published literature. Specifically, they champion a four-pronged approach to good experimental design: correctness (the use of good datasets), transparency, realism, and safety. After assessing a sample of 36 recent publications in the literature, they make the following claims:

²⁷ We use “replicate” in the sense distinguished by Nancy Cartwright (Cartwright 1991): to perform as closely as possible the experiment’s original procedures with the aim of acquiring the same results. Along with Alexander Rodriguez, we have been attempting to replicate the experiment performed by Yinglei Wang and colleagues presented in “Hiding Information in Flash Memory” (Wang et al. 2013). Even after acquiring from them the very device that they used in their original experiment, we have been unable to run the experiment from the information they provide in their publication. Subsequent correspondence with the authors for clarity also has failed to mitigate the problem. Further details may be presented in a future publication once the research is complete.

²⁸ See, for example, Maxion et al. (2010) and Killourhy and Maxion (2011).

²⁹ See Killourhy and Maxion (2009) for details.

³⁰ Because malware research heavily engages engineering, this case provides good evidence that even experimentation targeting engineered mechanisms in computing might benefit from adopting the methodology of established sciences. That stronger claim would require further argumentation that may be pursued in future work.

At least nine (25 %) distinct papers appear to suffer from clearly significant problems relating to our three most basic correctness criteria ... Descriptions of experimental setups lack detail to ensure repeatability ... [and] the majority of papers incompletely describe experimental results ... Only a minority of papers includes real-world evaluations, and very few papers offer significant sample sizes ... Most papers did not deploy or adequately describe containment. More than two thirds (71 %) completely omit treatment of any containment potentially used during the experiments. (Rossow et al. 2012, pp. 71–73)

They conclude that malware experimentation currently does not meet the acceptable standards of scientific respectability.

The four-pronged “prudence” approach to malware experimentation advocated by Rossow et al., reduces to careful assessment of the experimental mechanism. “Correctness,” or the use of good data sets, ensures that the results of the experiment conform to the “regularity” feature described by Machamer, Darden, and Craver: without good data sets, the experiment would not reveal a mechanism that works “the same way under the same conditions,” because the mechanism would be corrupted by a feature that would not necessarily manifest itself during a future experimental run. Correctness, therefore, is an instantiation of internal validity. “Transparency,” which lends repeatability and comprehensibility to the experimental setup, amounts to ensuring that no “explanatory gaps” exist in conveying the experiment’s design. This is in accord with the transparency desideratum described in Sect. 3.1. “Realism,” as described by Rossow et al., correlates directly with the external validity desideratum: it involves making sure that the experiment involves malware, operating systems, network presence, and stimuli that will accurately replicate situations that occur outside of the experimental mechanism. Lastly, “safety” involves controlling the set up and conclusion conditions of the experiment in such a way that the experimental mechanism cannot inadvertently be efficacious of (potentially harmful) consequences outside the experiment. That makes safety a correlate of the containment desideratum.

We see here a clear example of computing (in this case study, cybersecurity) potentially benefitting from adopting the methodology of an established science (in the case study depicted here, biology). We suspect that the example can be extended to other subfields of computing. For example, an inquiry that attempts to determine the scalability or stability of a system—whether it be the Internet, a large cloud infrastructure, or a consumer gaming platform—must deal with similar questions involving internal validity, external validity, containment, and transparency in order to be accurate, useful, and safe enough to conduct. Additionally, at a general level, the scientific reputability of the extant experimental research in computing comes into question largely due to experimental methodology.³¹ However, in opposition to the claim made by Schiaffonati and Verdicchio, the experimental methodology in computing *can* benefit from adopting the principles entrenched in the traditional sciences.

³¹ See, for example, Feitelson (2006), Freeman (2008), McCracken et al. (1979), Morrison and Snodgrass (2011), Vandewalle et al. (2009), Zerkowitz and Wallace (1997), Zerkowitz and Wallace (1998), Denning (1980), and Tichy (1998) for evidence that experimental shortcomings are the cause for computing’s lackluster scientific reputation.

As has been argued, adopting the mechanistic approach in conjunction with the methodology from established sciences may be beneficial for improving experimentation that isolates physical mechanisms in computing. Such adoption for experimentation on engineered mechanisms also may be fruitful. However, as will be described in Sect. 4, engineered mechanisms present special difficulties that warrant consideration from philosophers of science. Some of the researchers in cybersecurity are beginning to recognize the advantage of a mechanistic conception of experimental design, and the argumentation provided in this section has taken the first steps toward justifying the approach for computing at large.

4 Experimentation on Engineered Mechanisms in Computing

A fundamental assumption in the preceding section was that, when dealing with physical (natural) mechanisms, their operation is regular, consistent, and unchanging. This assumption is removed when dealing with engineered (artificial) mechanisms: owing to manipulation by rational agents, they may change quickly, with the same starting conditions leading to different results at the termination of the mechanism's operation. Technology's frequent use of engineered mechanisms marks a departure from the mechanisms under study in most established sciences. This makes worthwhile a cursory investigation into technology (and its experimentation on engineered mechanisms) from the perspective of philosophy of science. In this section, we adopt that perspective, while repeating once more that the following results represent very early work in exploring the implications for philosophers of science. To develop this facet of the project to fruition will warrant a separate, dedicated piece of research. For now, we merely turn to the initial impressions that might be the seeds of such future research.

Technology enables new action—new *activities*—that we have not witnessed before. Experimentation on engineered mechanisms often involves isolating the activities being performed by the entities in the mechanism, since the physical pieces of nature (the entities) are typically understood in such cases.³² The exploration thus involves understanding what those entities can *do*. This is complementary to the standard experimental approach in many physical sciences, such as physics, where the aim is to isolate and study the basic entities of the world.

For example, Allan Franklin has presented argumentation (on several occasions) for the different roles that experiment can play for scientists (Franklin 1981; 1990, Chapter 7; 2012, Section 2). By consolidating those accounts, we can generate the following list of the roles played by experiment:³³

³² An interesting question arises at this juncture concerning the ontological status of some components, such as software or algorithms, of engineered mechanisms. It is not readily apparent whether code (for example) should be classified as an entity or an activity, since a piece of code is referred to as an existing thing (i.e., entity), but it exists as such insofar as it has capacity to enact some change (i.e., activity).

³³ In recent personal communication with Allan Franklin, he suggested a few other roles: exploratory, calling for a new theory, measuring a quantity of physical interest, enabling, and pursuit. Some of these are subcategories of the listed roles, and others are still in the developmental stages of description, so we have restricted the items on the list to only those that feature prominently in the published works.

1. Theory choice
2. Theory articulation
3. Demonstration that entities involved in our accepted theories exist
4. Measurement of physical quantity
5. Life of its own

Franklin's list is not a description of stages involved in a single experimental process. Instead, it is intended to be a set of roles for experimentation whereby any particular experiment assumes one of the roles on the list. Theory choice involves confirming a theory, refuting a theory, choosing between theories, or calling for a new theory. Theory articulation involves stating an existing theoretical dictum in a new way, perhaps through a formula or law that is gathered from experimental practice. The demonstration that entities exist needs no further expounding. The measurement of physical quantities involves correcting misinterpreted results or achieving more accurate measurements, perhaps due to the acquisition of better equipment. The "life of its own" category is meant to capture the concept of exploratory work or "experimentation for experimentation's sake." It is experimentation that is undertaken independently of theoretical considerations.

Franklin's list suggests that philosophers of science tend to focus on the entities within a mechanism. Categories (1), (2), and (5) involve theoretical considerations. Only (3) and (4) directly address the experimental mechanism, and both exhibit emphasis on entities.³⁴

Particularly noteworthy with respect to Franklin's list is the inclusion of (3) in conjunction with the noticeable absence of its complementary role, which would be:

6. Demonstration that activities involved in our accepted theories occur.

This role is precisely the role played by many experiments in computing when the focus is an engineered mechanism. The experimenter wishes to determine whether some engineered mechanism will *act* a certain way.

For example, a cybersecurity researcher may wish to determine the mechanism by which compromised computers in a botnet communicate with their command and control infrastructure. An instance of the malicious software may be isolated in a laboratory environment to determine which activities it is capable of employing (direct IP connection, hard-coded domain name, domain generation algorithm, peer-to-peer, etc.). Observations must then corroborate which activity is used in actuality on the Internet. Further, the researcher may wish to refine the description of the activity so as to detect and isolate it reliably, such as the description of TDL4 botnet peer-to-peer communication described in Jerrim (2013).

Franklin's primary research program is experimentation in the physical sciences, so his emphasis on entities in experimental mechanisms might be expected. Indeed, by contrast, much experimental work in biology is tailored to discover the activities

³⁴ (4) could perhaps be construed as measuring an activity, but that interpretation would be questionable. Franklin's inclusion of the term "physical" suggests that entities are the pieces under measurement. Further, even if (4) applies to activities, it would refer to the measurement of already-understood activities, rather than understanding the activities themselves, as occurs in the case of technology.

involved in certain mechanisms. For example, two prominent mechanisms that feature in the piece by Machamer, Darden, and Craver are the mechanism by which chemical transmission occurs and the mechanism by which protein synthesis occurs (Machamer et al. 2000, pp. 8–16). In both cases, the entities involved are broadly understood, but the task is to understand *how the mechanism acts* through completion. This further likeness between computing and biology—both include some experimental mechanisms for which the goal is understanding the mechanism’s activities—further strengthens the result from Sect. 3. Computing is closely akin to established sciences in certain ways, so adopting some experimental methodology from those sciences can be beneficial.

Additionally, however, the reverse has now surfaced: some established sciences are akin to computing in certain ways, so philosophers of science might stand to benefit from investigating the likeness. The preceding suggests that philosophers of science might obtain methodological clarity by splitting the roles for experiment accordingly and including (6) among the roles. Analysis of engineered mechanisms in computing may reveal further work for philosophers of science, as well. The intrinsic susceptibility of engineered mechanisms to change might offer such an opportunity. In that case, the analysis might, for example, result in the emergence of a desideratum specifically addressing the acceptable level of manipulation (if any) of the engineered mechanism under experimental consideration. In these ways (and almost certainly others, as well), the “regularity” (or lack thereof) of engineered mechanisms offers philosophers of science the possibility for novel research opportunities.³⁵

The established mechanistic approach includes an ontology of entities and activities. The aim of a properly constructed scientific experiment is to isolate either an entity or an activity within a mechanism so that it can be identified or understood. Attempting to do more within a single experiment introduces confounding factors that may cast doubt upon the experiment’s results. Further analyses of the mechanistic approach to experimentation in computing, in addition to a deeper analysis of the implications that have been suggested in this section, might reveal additional hitherto unidentified roles for experiments, but such analyses must be left to future research.

5 Concluding Remarks

A promising hint of correspondence emerges from the preceding narrative. Mechanisms are comprised of entities and activities. Computing admits of experimentation on physical mechanisms and engineered mechanisms. A mechanistic approach to experimentation in computing shows that researchers in computing stand to benefit by adopting methodology from established sciences, especially when the experimentation targets a physical mechanism. Likewise, philosophers of science stand to benefit from studying the implications of the mechanistic approach for experimentation in computing. Experimentation in some sciences admits of the same conceptual split between entity-focused experimentation and activity-focused experimentation that is inherent in

³⁵ We thank an anonymous referee for suggesting the difficulty encountered by an experimenter when faced with the potential irregularities that might emerge while designing an experiment that targets an engineered mechanism.

computing. Accordingly, detailed study of experimentation on engineered mechanisms, which often involves an activity-focused approach, could be fruitful for philosophers of science.³⁶

Thus, the argumentation provided by Schiaffonati and Verdicchio pays dividends. Refining their “engineering” insight shows that computing often involves *engineered* mechanisms. Targeting the activities exhibited by such mechanisms is, descriptively, what many researchers undertake when experimenting. That goal is no less a scientific undertaking than, for example, the experimentation performed by biologists in their attempts to understand the activities involved in chemical transmission in synapses. The mechanistic approach to experimentation in computing provides significant evidence that computing belongs among the other scientific endeavors. Addressing the potential practical difficulties involved in adequately describing an engineered mechanism, which may change while the experimenter is investigating, may be the next step. However, computing does not have a completely different conceptual framework from other sciences. Computing can benefit from a mechanistic approach to experimentation.

Acknowledgments This material is based upon work funded and supported by the Department of Defense under contract no. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. This material has been approved for public release and unlimited distribution. Carnegie Mellon®, CERT®, and CERT Coordination Center® are registered marks of Carnegie Mellon University. DM-0000768

Open Access This article is distributed under the terms of the Creative Commons Attribution License which permits any use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

References

- Cartwright, N. (1991). Replicability, reproducibility, and robustness: comments on harry collins. *History of Political Economy*, 23(1), 143–155.
- Copeland, B. J. (1996). What is computation? *Synthese*, 108(3), 335–359.
- Denning, P. J. (1980). What is experimental computer science? *Communications of the ACM*, 23(10), 543–544.
- Dhamija, R., Tygar, J. D., & Hearst, M. (2006). *Why phishing works*. New York, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI’06).
- Feitelson, D. G. (2006). Experimental computer science: the need for a cultural change. [Online] Available at: <http://www.cs.huji.ac.il/~feit/papers/exp05.pdf>
- Franklin, A. (1981). What makes a ‘good’ experiment? *The British Journal for Philosophy of Science*, 32(4), 367–374.
- Franklin, A. (1990). *Experiment, right or wrong*. Cambridge: Cambridge University Press.
- Franklin, A. (2012). Experiment in physics.[Online] Available at: <http://plato.stanford.edu/archives/win2012/entries/physics-experiment/>.
- Freeman, P. A. (2008). Back to experimentation. *Communications of the ACM*, 51(1), 21–22.
- Fresco, N. (2013). Information processing as an account of concrete digital computation. *Philosophy & Technology*, 26, 31–60.
- Fresco, N., & Wolf, M. J. (2013). The instructional information processing account of digital computation. *Synthese*, 191, 1469–1492.
- Friedl, S. (2008). *An illustrated guide to the Kaminsky DNS vulnerability*. [Online] Available at: <http://unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html>. Accessed 10 Jan 2014.

³⁶ We do not mean to suggest that experiments on engineered mechanisms in computing *only* involve focus on activities. Some such experiments involve the isolation of entities.

- Gagliardo, A., et al. (2013). Oceanic navigation in Cory's shearwaters: evidence for a crucial role of olfactory cues of homing after displacement. *The Journal of Experimental Biology*, 216, 2798–2805.
- Glennan, S. (2014). Mechanisms. In M. Curd & S. Psillos (Eds.), *The Routledge companion to philosophy of science* (2nd ed., pp. 420–428). New York: Routledge.
- Jerrim, J. (2013). *Detecting malware P2P traffic using network flow and DNS analysis*. Albuquerque: FloCon.
- Killourhy, K. S., & Maxion, R. A. (2009). *Comparing anomaly-detection algorithms for Keystroke dynamics* (pp. 125–134). s.l., s.n.
- Killourhy, K. S., & Maxion, R. A. (2011). *Should security researchers experiment more and draw more inferences?* (pp. 1–8). s.l., s.n.
- Machamer, P., Darden, L., & Craver, C. F. (2000). Thinking about mechanisms. *Philosophy of Science*, 67, 1–25.
- Maxion, R. A., Longstaff, T. A., & McHugh, J. (2010). *Why is there no science in cyber science?: a discussion panel at NSPW 2010* (pp. 1–5). s.l., s.n.
- McCracken, D. D., Denning, P. J., & Brandin, D. H. (1979). An ACM executive committee position on the crisis in experimental computer science. *Communications of the ACM*, 22(9), 503–504.
- Morrison, C. T., & Snodgrass, R. T. (2011). Computer science can use more science. *Communications of the ACM*, 54(6), 36–38.
- Piccinini, G. (2007). Computing mechanisms. *Philosophy of Science*, 74(4), 501–526.
- Piccinini, G., & Scarantino, A. (2011). Information processing, computation, and cognition. *Journal of Biological Physics*, 37, 1–38.
- Radder, H. (Ed.). (2003). *The philosophy of scientific experimentation*. Pittsburgh: University of Pittsburgh Press.
- Reade, W. C., Ellingson, D. L., & Lindsay, J. (2007). *Jamming device against RFID smart tag systems*. United States of America, Patent No. 7,221,900.
- Rosow, C., et al. (2012). *Prudent practices for designing malware experiments: status quo and outlook* (pp. 65–79). s.l., s.n.
- Scheutz, M. (1999). When physical systems realize functions... *Minds and Machines*, 9, 161–196.
- Schiaffonati, V., & Verdicchio, M. (2013). Computing and experiments: a methodological view on the debate on the scientific nature of computing. *Philosophy & Technology*.
- Shenghua, Z., Dazhuan, X., Xueming, J., & Hua, H. (2004). A study on active jamming to synthetic aperture radar. *3rd International Conference on Computational Electromagnetics and Its Applications* (pp. 403–406).
- St. Sauver, J. (2009). *Electromagnetic pulse (EMP)*. Eugene, Oregon, Infraguard Meeting.
- Tichy, W. F. (1998). Should computer scientists experiment more? *IEEE Computer*, 31, 32–40.
- Vandewalle, P., Kovacevik, J., & Vetterli, M. (2009). Reproducible research in signal processing. *IEEE Signal Processing Magazine*, 26, 37–47.
- Wang, Y., et al. (2013). *Hiding information in flash memory*. s.l., s.n.
- Weber, M. (2012). *Experiment in biology*. [Online] Available at: <http://plato.stanford.edu/archives/spr2012/entries/biology-experiment/>.
- Zelkowitz, M. V., & Wallace, D. (1997). Experimental validation in software engineering. *Information and Software Technology*, 39, 735–743.
- Zelkowitz, M. V., & Wallace, D. (1998). Experimental models for validating technology. *IEEE Computer*, 31, 23–29.