Computation and levels in cognitive and neural science

Lotem Elber-Dorozko and Oron Shagrir

Introduction

Many neuroscientists, cognitive scientists and philosophers take it that nervous systems, and even their single cells, perform computations and that these computational operations play a role in producing and explaining cognition. Scientists and philosophers also tend to account for neuro-cognitive phenomena within one or another framework of levels.

How are these two trends – the computational approach and the framework of levels –integrated in the foundations of cognitive science? The answer to this question is by no means simple, and largely depends on what one means by *levels* and by *computation*. Leveled hierarchies have been used to present various, sometimes conflicting, viewpoints. While some have presented such hierarchies to argue for the reduction of all levels to one fundamental level (Oppenheim and Putnam, 1958), others have used them to support the autonomy of different scientific practices (Cummins, 1983), still others presented them as a comprehensive way to explain cognitive phenomena (Marr, 1982). Hence, Levels can be understood as levels of description (Pylyshyn, 1984), analysis (Marr, 1982), organization (Churchland and Sejnowski, 1992), mechanisms (Bechtel, 1994; Craver, 2007), and more.[1]

The nature of computation and computational explanation is also a hot topic, with answers spanning semantic (Shagrir, 2006; Sprevak, 2010), syntactic (Stich, 1983), causal (Chalmers, 2011), mechanistic (Milkowski, 2013; Piccinini, 2015), algorithmic (Copeland, 1996) and other approaches. We will discuss here three approaches to computational explanations and how they fit in a framework of levels. One is David Marr's (1982) three-level analysis in which the top-level consists of computational-level theories. Another is offered by Cummins (1983, 2000); see also (Fodor, 1968; Haugeland, 1978), who proposes that computational explanations are a sort of functional analysis, and that they are autonomous to some extent from the lower implementation (e.g., neurological) level. A third, more recent, picture views computational explanations as

[1] See Craver (2007, chapter 5) for a review.

mechanistic explanations (Piccinini 2007; 2015; Piccinini and Craver 2011; Kaplan 2011; Milkowski 2013; Fresco 2014) that are not autonomous from a lower implementation level.

The chapter does in no way provide an exhaustive survey of all the literature concerning computation and levels; nor does it offer a systematic analysis of the relations between them. Our aim is far more modest: We focus on on-going debates concerning the place of computational explanations within a leveled approach to the study of neuro-cognitive phenomena. As a teaser, we start with a quick mention of three influential pictures of levels that had their influence on the pictures described in more depth along the chapter.

We begin with a picture that deals with scientific practice in general, but greatly influenced frameworks for explanation in cognitive science. Oppenheim and Putnam (1958) discuss levels of scientific discourse, each employing its own predicates and laws (e.g., atoms, molecules, living things). The levels are hierarchically ordered such that "Any thing of any level except the lowest must possess a decomposition into things belonging to the next lower level" (p.9). Oppenheim and Putnam use this framework to argue that, based on empirical evidence, it is very likely that there is a "unity of science". This means that each level can be reduced into its next lower level so that any observational data explainable by the former is explainable by the latter. Because reduction is transitive, all levels can be reduced into one united fundamental level. "In this sense, Unity of Science is an alternative to the view that it will eventually be necessary to *bifurcate* the conceptual system of science, by the postulation of new entities or new attributes.." (p.13).

A second picture, that deals specifically with explanation of cognition, is offered by Zenon Pylyshyn (1980, 1984, 1989), and is consonant with the framework offered by Allen Newell (1980, 1982).[2] Pylyshyn advances a tri-level framework with the aim of providing foundations for the classical view in cognitive science.[3] Assuming that cognition is classical (e.g., digital and programmed) computation, it would be best to account for cognition at different levels, wheras each level describes the same system/process from a different perspective. Pylyshyn enumerates "three autonomous levels of description " (1984, p. 259). The top level is the semantic (or knowledge) level. This level explains why people, or other complex computing systems, do certain

[2] The term knowledge level arrives from Newell (1982) who also terms the second level "symbolic".
[3] In the earlier writings, Pylyshyn (1984) talks about cognitive science, but later on, perhaps with the rise of the rival connectionist approach, he confines the proposed tri-level framework to the classical view (1989, p. 57).

things by stating their goals as well as "showing that these are connected in certain meaningful or even rational ways" (1989, p. 57). The symbol (or syntactic, algorithmic or functional) level demonstrates how the semantic content is encoded by symbolic expressions; it also specifies the structure of these expressions and the regularities by which they are manipulated. The physical (or biological) level specifies how the entire system is realized in physical structures.[4] While each level describes different aspects of a computing system, Pylyshyn argues that the distinctive feature of the computational approach is that it addresses cognition in terms of a formal symbolic, algorithmic level of analysis (1980, p. 111).[5] The computational level is not detached from the semantic and the physical levels; it constrains and is constrained by them. The levels are autonomous in the sense that each addresses different questions and conforms to different principles (1984, p. 259).

A very different picture is offered by Patricia Churchland and Terrence Sejnowski (1992) in *The Computational Brain*. Their starting point is "that there are levels of organization is a matter of fact" (p. 11). When talking about "levels of organization", Churchland and Sejnowski take it, similarly to Oppenheim and Putnam (1958), that objects belonging to a lower-level are smaller in size, and are also parts of objects that are at the higher level (Fig. 1). This is in contrast to Pylyshyn who often talks about levels of description in which the entities at a lower level (e.g., symbolic) are very often not part of entities at the higher level (e.g., semantic content).[6] Another difference is that most or all the levels mentioned by Churchland and Sejnowski (Fig. 1) fall within Pylyshyn's physical/biological level. Moreover, while Churchland and Sejnowski certainly associate computing with information-processing, they do not talk about distinct semantic or functional levels. As they see it, computational studies contribute to the "co-evolution" of research in different levels. This approach, they bet, is key to the understanding of brain and cognitive function. While Pylyshyn’s view, levels of description, is reflected in Cummins’ and to lesser extent in Marr's

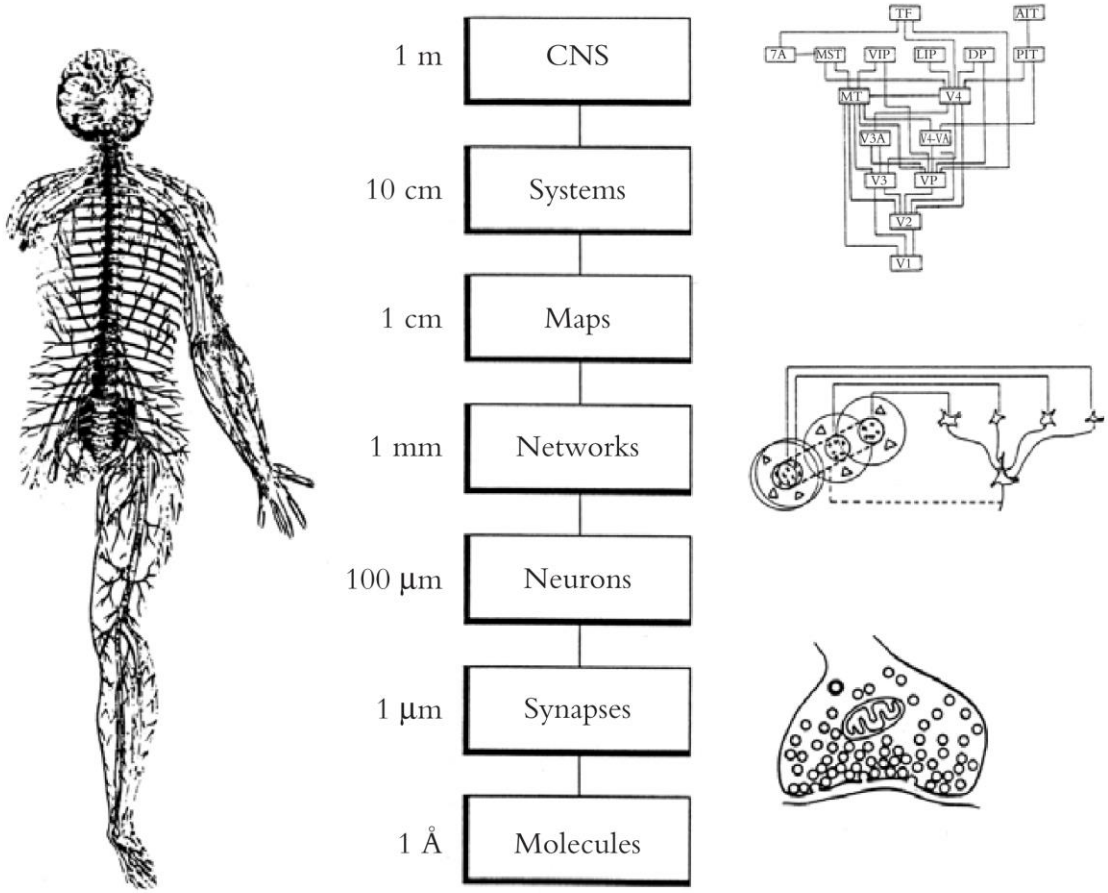
[4] This tri-level picture bears similarities to Dennett's tri-stance picture (Dennett 1971), which offers to account for the behavior of complex systems. The semantic level parallels in some respects to the intentional stance. The design stance has some affinities with the functional aspects of Pylyshyn's symbolic level, and the physical stance is similar to the physical level.

[5] As can also be seen in Section 2, the "computational levels" are often contrasted with the physical levels, where the computational levels are identified with functional and semantic levels (Block, 1995), and sometime exclusively with a syntactic level (Fodor, 1994, pp. 7–16).

[6] Some take levels of organization to be levels of nature that describe leveled objects in the world, while levels of description are levels of science that describe the relations between scientific theories or models (Craver, 2007, chap. 5). We do not discuss this distinction in this chapter.

writings (Section 1 and 2), the mechanistic approach adopts levels with part-whole relations, similarly to the “levels of organization” view (Section 3) and the tension between these different frameworks is the basis for many debates today.

Figure 1 (adapted with permission from Churchland and Sejnowski; 1988) – Levels of organization in the nervous system.



1. Marr's computational-level theories

In *Vision*, Marr famously advances a three-level approach to the study of visual processes and to the study of cognition more generally. The "most abstract" is the computational level (CL), which "is the level of *what* the device does and *why*" (p. 22). The role of the what-aspect is to specify what is computed. The job of the why-aspect is to demonstrate the appropriateness and adequacy of what is being computed to the information-processing task (pp. 24-25). We will discuss the computational level in more detail below. The algorithmic level characterizes the system of

representations that is being used, e.g., decimal vs. binary, and the algorithm for the transformation from input to output. The implementation level specifies how the representations and algorithm are physically realized. Marr's levels are not levels of organization, where the entities at higher level are composed of entities at the level below it. He refers to his levels as "levels of analysis", whereas each such level provides a further understanding of the visual phenomenon. Marr's levels are similar to Pylyshyn's levels in that each level employs its own methods to analyze the cognitive system, and in that the interaction between the (distinct) levels is essential for a complete account of the explanandum phenomenon, e.g., some visual task. However, some identify Marr's 'computational level' with Pylyshyn's semantic level, and Marr's *algorithmic level* with Pylyshyn's *symbolic level*. We challenge this view and will elaborate on it more below.

Marr says that "it is the top level, the level of computational theory, which is critically important from an information-processing point of view" (p. 27), and he distinguishes it from the algorithmic and implementational levels that deal with the underlying processes at our heads:

> There must exist an additional level of understanding at which the character of the information-processing tasks carried out during perception are analyzed and understood in a way that is independent of the particular mechanisms and structures that implement them in our heads. This was what was missing – the analysis of the problem as an information processing task (p. 19).

Marr, however, never provided a systematic and detailed account of his notion of CL. He moves on to advance a set of computational theories of specific visual tasks which had a tremendous impact on vision research. Explicating the notion of a computational-level theory and its place in the level-picture was left to philosophers, who provided, in turn, very different interpretations. We will review some of these interpretations. But we want to emphasize that our motivation is not interpretative, namely providing the most faithful interpretation of Marr. Our aim is to highlight an important feature of computational explanations that is (arguably) captured by Marr's notion of CL.[7]

On the "standard" interpretation, the role of CL is to specify the cognitive phenomenon to be explained: "A computational analysis will identify the information with which the cognitive

[7] See Shagrir and Bechtel (2017) for a detailed discussion of these interpretations.

system has to begin (the *input* to that system) and the information with which it needs to end up (the *output* from that system)" (Bermúdez, 2005, p. 18). Thus edge-detection is the mapping from representations of light intensities to representations of physical edges (e.g., object boundaries). Shape-from-shading is the mapping from representations of shading to representations of shape, and so on. The explanation itself is then provided mainly at the algorithmic level (Ramsey, 2007, pp. 41–42). This interpretation is inspired by Pylyshyn's three-layer picture on which the top-level is some "semantic" level, and it is the middle, symbolic, level, that explains how the system performs cognitive capacities. But it is not completely in accord with Marr who separates the computational and algorithmic levels and assigns to the computational level a unique explanatory role.[8] The standard interpretation is right in saying that Marr's computational level can be seen as delineating the explanandum for the algorithmic and mechanistic levels. What is ignored, however, is the fact that this delineation has itself a major explanatory role in the account of the visual task (Shagrir and Bechtel, 2017).

Lawrence Shapiro (1997) interprets CL as providing a task analysis of the visual task. He writes that "at the computational level of theory the theorist describes what I shall call *chief* tasks and *service* tasks… the chief task of the visual system is the derivation of 3-D shape representations from information encoded in 2-D images. Service tasks are those tasks the completion of which contribute to the achievement of the chief task" (p. 134). In particular, he argues, the information-processing description of the service tasks, in terms of informational content, contributes to the understanding of the chief task. This is in accord with the functional picture of computational explanations – discussed in the next section – according to which the capacity of a system ("chief task") is explained in terms of the capacities of the components ("service tasks") of which it is composed. According to Shapiro, only after this task is completed, we turn to the algorithmic level that specifies "the algorithms and representations that can in fact solve the chief and service tasks" (p. 136).

There are indeed cases where the information-processing descriptions of service tasks (e.g., stereo disparity) account for the chief task (e.g., stereo vision). But the paradigm cases of computational theories that Marr and his students advance – edge-detection, stereo disparity, and structure-from-

[8] No wonder that these interpreters have commented that "Marr, very confusingly, calls it [the top level] the 'computational' level" (Sterelny, 1990, p. 46). Dennett (Dennett, 1994, p. 681) and Ramsey (2007, p. 41 note 43).

motion – do not aim to provide task analyses. For example, the theory of stereo disparity demonstrates that the system computes a matching function that must satisfy the constraints of uniqueness and continuity (these constraints reflect, in turn, certain facts about the visual field). In many cases, the functional strategy proposed by Shapiro better fits with Marr's algorithmic level than with his computational level. Thus, while the computational level aims to explain what the system is computing and why it is computing it, a decomposition of the task will usually explain how the task is achieved and such a description is more in accord with a description of an underlying algorithm.

Gualtiero Piccinini and others (Piccinini and Craver 2011; Piccinini 2015; Boone and Piccinini 2016b; see also Kaplan 2011, p. 343) describe Marr's computational and algorithmic levels as *sketches* of mechanisms (see Section 3 on mechanisms). A sketch of a mechanism is a description in which some structural aspects of the mechanism are missing. Once these missing aspects are filled in, the description turns into "a full-blown mechanistic explanation"; the sketches themselves can be thus seen as "elliptical or incomplete mechanistic explanations" (Piccinini and Craver, 2011, p. 284). They are, in a way, a guide or a first step towards the structural components that constitute the full-blown mechanistic explanations. Piccinini would probably agree with Shapiro that the computational (and algorithmic) level provides a task analysis, but he argues that a task analysis is a mechanistic sketch (Piccinini and Craver, 2011), and computational explanations are mechanistic explanations (Piccinini, 2015).

Piccinini is right to observe that both the computational and algorithmic levels are abstract, in that they omit certain structural aspects of the mechanism (both levels are also abstract in the sense that they provide mathematical or formal descriptions). We disagree, however, that these levels provide weak or incomplete explanations (See Section 3). What seems even more troublesome is the attempt to lump Marr's computational and algorithmic levels together. If anything, it is the algorithmic and implementational levels that belong together as both look *inside* the computing system to the operations that enable it to compute a function. The algorithmic level (much like the implementation level) is directed to the *inner working* of the system, i.e. to causal relations between sub-components. In contrast, the computational level looks *outside*, to identifying the computed function and to relating it to the environment in which the system operates (Shagrir and Bechtel, 2017).

Frances Egan (1995, 2010, 2017) argues that CL provides a mathematical specification of the input-output mathematical function that the system computes (then the algorithmic level specifies the algorithm by means of which the system computes this function, and the implementation level specifies how this algorithm is implemented in the brain): "The top level should be understood to provide a function-theoretic characterization", and "the theory of computation is a mathematical characterization of the function(s) computed" (Egan, 1995, p. 185). Thus, for example, the computational theory of early vision provides the mathematical formula $\nabla^2 G * I$ as the computational description of what the retina does.[9] This, according to Egan, is an explanatory formal theory.[10] It is not, however, a representational theory: "*Qua* computational device, it does not matter that input values represent *light intensities* and output values the rate of change of *light intensity*. The computational theory characterizes the visual filter as a member of a well understood class of mathematical devices that have nothing essentially to do with the transduction of light" (Egan, 2010, p. 255). The cognitive, intentional, characterization is what Egan terms a *gloss* on the mathematical characterization provided by the computational theory; but it is not part of the theory.

Egan nicely captures the way Marr characterizes the *what* aspect of CL. The job of this element is to provide a precise specification of *what* the system does, and the precise specification of what the retina does is provided by the formula $\nabla^2 G * I$. However, Egan downplays the fact that there is another component to CL, namely, the *why* aspect. So it seems that Marr thinks that CL has to cover another aspect, beyond providing mathematical characterizations.

A more recent interpretation emphasizes the role of the environment in Marr's notion of computational analysis (Shagrir, 2010; Bechtel and Shagrir, 2015; Shagrir and Bechtel, 2017). According to this interpretation, the aim of the *what* element is to characterize the computed (typically input-output) function in precise mathematical terms. The aim of the *why* is to demonstrate why the computing function is pertinent to the visual task. Thus to take the theory of

[9] The term I stands for a two-dimensional array ("the retinal image") of intensity values detected by the photoreceptors (which is the input). This image is convoluted (here signified by '$*$') through a filter $\nabla^2 G$, where G is a Gaussian and ∇^2 is a second-derivative (Laplacian) operator. This operation is arguably performed in the retinal ganglion cells.

[10] A similar viewpoint is expressed by van Rooij (2008). An important variant of this view associates the computational level with an idealized *competence* and the algorithmic and implementation levels with actual performance (Horgan and Tienson, 1994; Frixione, 2001; Polger, 2004; Rusanen and Lappi, 2007).

edge-detection, the *what* element characterizes the operations of early visual processes as computing the zero-crossings of (Laplacian) second derivative filterization of the retinal images. The aim of the *why* is to say why the visual system computes derivation, and not (say) factorization or exponentiation, for the task of edge-detection. Saying that the computed function leads to representations of edges just reiterates the *why* question. After all, computing derivation in very different environments – where sharp changes in illumination conditions occur very frequently or along the solid faces of surfaces – would not lead to representations of edges. So why is derivation pertinent to edge-detection in our visual environment?

Marr associates the *why* aspect with what he calls *physical constraints*, which are physical facts and features in the physical *environment* of the perceiving individual (1982, pp. 22–23). These are constraints in the sense that they limit the range of functions that the system could compute to perform a given visual task successfully. In the case of edge-detection one relevant constraint is that sharp changes in reflection (e.g., illumination conditions) often occur along physical edges such as object boundaries. Thus by detecting the zero-crossings of the second-derivative operators the visual system captures the sharp changes in light reflection in the visual field (whereas the latter changes can be described in terms of extreme points of first-derivatives or zero-crossings of second derivatives of the reflection function). The claim, more generally, is that a computational analysis appeals to the physical constraints in order to underscore some structural similarities (or morphism) between the visual systems and the visual field, and these, in turn, serve to demonstrate the appropriateness and adequacy of the computed function to the information-processing task.

To sum up, Marr's notion of CL has provoked a variety of interpretations which reflect, to large extent, the views of the interpreters about the nature of computational explanations and levels in cognitive science. Often, these interpretations assimilate the computational level with the algorithmic (e.g., Ramsey and Shapiro) and even with the implementation (e.g., the mechanistic view) level. We take these views to miss an important aspect of CL, namely, how the computation reflects the environment. Nonetheless, these views have their own merits. In the next two sections we will discuss two of these views (the functional and the mechanistic) in some more detail.

2. Computational explanation as functional analysis

An influential position in philosophy of cognitive science, advocated mostly at the end of the twentieth century, is that cognitive capacities are explained by appeal to (simpler) cognitive functions and their interaction (Fodor, 1968; Haugeland, 1978; Cummins, 1983, 2000). For example, the "capacity to multiply 27 times 32 analyzes into the capacity to multiply 2 times 7, to add 5 and 1, and so on" (Cummins, 2000, p. 126). According to this position there are two levels at which a system can be described: the functional level and the physical level. The functional level, addressed in psychological research, describes properties and activities as having a functional-teleological role, often the states and activities are presumed to have some intentionality, e.g., activities are described as computations, information-processing tasks and semantic tasks. The physical level, addressed in neuroscience, describes physical structures and properties, without a functional interpretation. On this view, computational explanations, that describe computing functions, are functional analyses that reside at the functional level because they analyze the capacity and not the realizing system. As Cummins writes: "Turing machine capacities analyze into other Turing machine capacities. Since we do this sort of analysis without reference to a realizing system, the analysis is evidently not an analysis of an realizing system but of the capacity itself." (2000, p. 125).

The functional and structural levels are levels of description, where the same system is described from different viewpoints. The functional and structural descriptions are taken to be autonomous and distinct from one another and it is usually stated that both are required for a complete explanation of the phenomenon (Fodor, 1968; Cummins, 1983). This position is radically different from the, then very popular, philosophical view advocated by Oppenheim and Putnam (1958); Oppenheim and Putnam argued that as science advances we expect types and theories in all scientific fields to be reduced to types and theories in physics. Therefore, phenomena that are explained today by theories that are outside the field of physics will someday be explained by a theory in physics. In contrast, proponents of the functional analysis view rely on the unique intentional features of cognition to suggest that science will and should remain divided to (at least) two levels that use different, unique explanations: the functional and structural levels.

In support of the claim that functional analysis is distinct from structural description, it is common to invoke Leibniz' gap. One argues that even if we had all the information about brain processes,

specified in neurological terms, we still could not deduce from it which functions are computed at the functional level. Also, when functionally analyzing a capacity, it is explained by decomposing it to simpler functions. To give an example of a computational explanation, choosing the next move in a chess game can be decomposed into simpler computations such as: computing the value of each state of the board, computing the probability of the board being in each state several moves from now, etc. These computations in turn are explained by decomposing them into simpler functions. But no matter how simple, the explaining activities are always also described functionally. In order to explain a function in structural rather than functional terms a different, non-functional, explanation is required (Haugeland, 1978; Cummins, 2000). Thus, although there are part-whole relations in functional analysis, according to this view these parts and wholes remain at the same, functional, level and what differentiates the levels is the type of description (functional vs structural). In this, this view follows Pylyshyn (1984, 1989), who presents a framework where different levels of description differ according to the aspects of the system they describe, and differs from Churchland and Sejnowski (1992), that describe levels of organization where the levels stand in part whole relations.

In support of the claim that the functional level is autonomous from the structural level, it is common to invoke multiple realization. Several people have argued, against the reductionist view (Oppenheim and Putnam, 1958), that it is extremely improbable that a specific type of mental state could be realized only by a single physical type. It is much more likely that mental properties have multiple possible physical realizations and therefore cannot be reduced to a single physical type (Putnam, 1967; Fodor, 1974). In a similar vein, it is indicated that functions and computations are often actually realized in different physical media. For example, both an eighth-grade student and an electronic calculator can subtract numbers, but the physical systems realizing the computation are different. One might conclude that "functional analysis puts very indirect constraints on componential analysis" (Cummins, 1983, p. 29, 2000, p. 126) and the practice of functional analysis of the mental can continue without paying much attention to the underlying neurology. On this framework science is bifurcated into the investigation of the functional and the investigation of the physical, where each can continue separately. .

While this picture is similar to Marr's framework in that the levels are levels of description, with computational descriptions at the top levels, this view interprets computational explanations as

explanations that decompose functions into other functions and it does not address Marr's emphasis on explaining *what* and *why* at the CL. As stated in Section 1, computational explanations as functional analyses are akin to Marr's algorithmic level. Moreover, Marr presented an integrative framework for explanation, whereas this view emphasizes the autonomy and distinctness of explanations at the functional and structural levels.

This disunity of science has been criticized by Lycan (1987, chap. 4). Lycan argues that functional and structural phenomena are not divided into two levels. According to Lycan, there are multiple levels, so that upper level types are composed of lower level types and explained by them. Functional descriptions become more structural as we descend down the levels. He illustrates this by referring to a possible functional analysis of a *face recognizer* (1987, pp. 43–44). According to this example, one sub-function that composes the *face recognizer* is an *analyzer*, which takes the visual picture as input and returns a binary vector computed from the picture. This *analyzer*, in turn, consists of a *projector*, that projects a grid on the picture and a *scanner* that runs through the squares and returns '1' or '0' for each square. One of the sub-functions that compose the *scanner* is a *light meter* that registers the degree of lightness in a square, which in turn is composed of *photosensitive chemicals*, and so forth. Lycan asks: "Now at what point in this descent through the institutional hierarchy (from *recognizer* to *scanner* to *light meter* to *photosensitive substance*…) does our characterization stop being teleological, period, and start being purely mechanical, period? I think it is clear there is no such point…" (1987, p. 44). Under this framework, a functional analysis of a capacity takes us one step closer to a structural description.

More recently, Piccinini and Craver (2011) have taken a stance similar to Lycan and argued that functional analyses are elliptical mechanistic explanations. Similar to functional analyses, mechanistic explanations (see also Section 3) explain phenomena by appealing to components and their interaction. Unlike functional analyses, they include details about the structure of the components. Piccinini and Craver argue that functional analyses should include structural information as well because, regardless of whether the functional is multiply realized in the structural, the functional directly constrains the structural and vice versa. As an illustration they write: "stirring (a sub-capacity needed in cooking) is the manifestation of (parts of) the cook's locomotive system coupled with an appropriate stirring tool as they are driven by a specific motor program" (2011, p. 293). The functional, in this case stirring, constrains the structural, in this case

an appropriate stirring tool. ” If the cook lacks a stirring tool but still manages to combine ingredients thoroughly, we should expect that the mixture has been achieved by other means, such as shaking” (2011, p. 293). Therefore, to know whether a functional analysis is indeed a true explanation of a capacity we must know at least that the postulated functions can take place in the structure of the brain (This is also pointed out by proponents of functional analysis, see Cummins 2000; Fodor 1968).

Both Lycan (1987) and Piccinin and Craver (2011) contest the claim that there are two autonomous levels of description and instead suggest a picture where levels are levels of organization (recall (Churchland and Sejnowski, 1992)) that stand in part-whole relations. On this picture both functional and structural descriptions persist throughout the levels in varying degrees,

The debate about the autonomy of functional analyses is still ongoing. Shapiro (2016) disputes Piccinini and Craver’s claims and argues for the autonomy of functional description from structural description. He points out that practically every explanation is constrained to some extent by physical details and that this should not be sufficient to threaten the autonomy of a functional explanation. He argues that while details about implementation are useful in supporting theories, they are not the only kind of evidence used to this effect; behavioral experiments are frequently used. Therefore, explanations in psychology can be shown to describe the actual causal structure without appeal to implementation details (see also Weiskopf 2011). Moreover, even when implementation details are used to support a theory, their confirmatory role does not make them part of the explanation itself, which may be completely functional.

This being said, scientific practice today clearly favors an integrated approach, as demonstrated in cognitive-neuroscience. Today, a wide variety of methods allows scientists to investigate neural activity in various time and space scales; such methods include single-unit and multi-unit electrophysiology, calcium imaging, fMRI, gene-expression profiling and more. Other methods such as optogenetics and gene trapping allow scientists to intervene on the activity of specific neurons and genes and observe the behavioral results. It is not surprising to see that many publications today relate cognitive phenomena with neural activity, sometimes at the level of gene transcription or spine growth. To name just a few examples, inhibition of dorsal subiculum neurons during recall has been shown to impair long-term memory in certain tasks (Roy *et al.*, 2017); ‘grid cells’ that fire when the animal is in specific locations in a grid-like manner were identified in the

entorhinal cortex (Fyhn *et al.*, 2004); and a large family of at least one hundred genes (today estimates are at 1000) has been found to encode proteins that are odor receptors, responsible for odor recognition (Buck and Axel, 1991).

Furthermore, as knowledge about brain structures expands, it is becoming more and more common to base computational models on known anatomical structures in the brain. One such example is a 'covariance based plasticity' learning algorithm, where learning is mediated through plasticity in the synapses of a network model (Loewenstein and Seung, 2006). Another famous example are deep-learning networks, which have been inspired by the hierarchical nature of the visual cortex. Often, activity in simulated neurons in these networks is compared with activity in visual brain areas (Lee, Ekanadham and Ng, 2007).

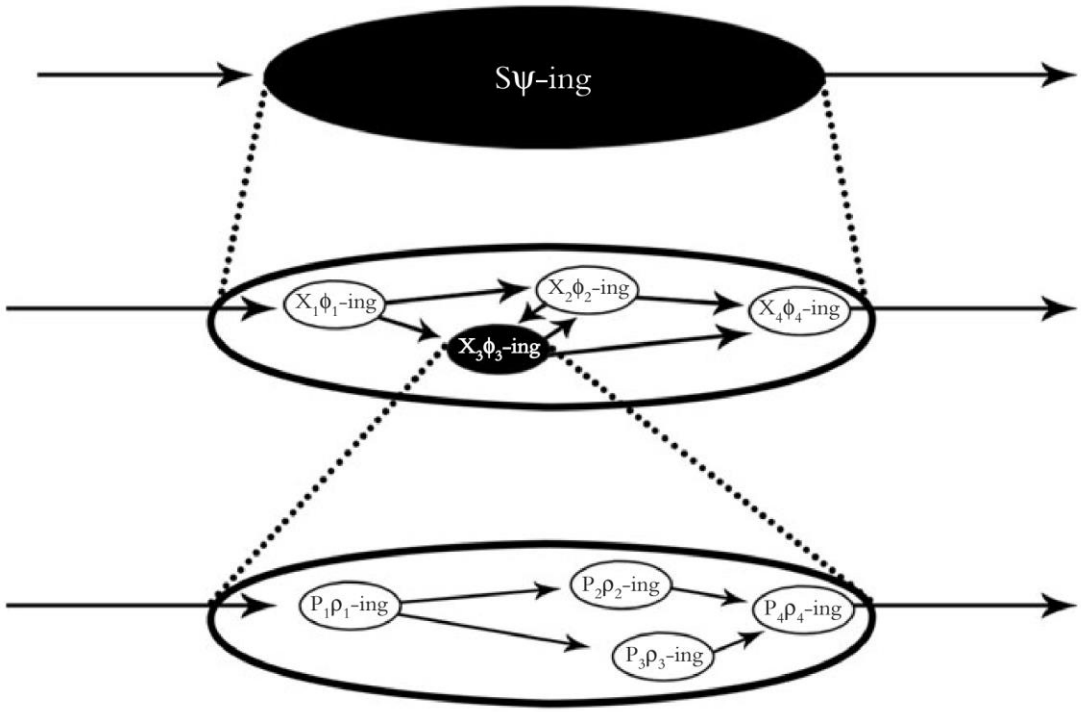
Much is still unknown about the complex relation between structure and function in the brain and those who point at possible ways to integrate the structural and functional do so with caution. Nonetheless, it is clear that today the study of the functional is not moving forward independently from the structural. On the contrary, much scientific effort today is dedicated to the investigation of their relation. This calls for a framework that integrates computational-functional properties and structural properties (Boone and Piccinini, 2016b). Such a view will be presented in the next section.

3. The mechanistic approach to computation

The mechanistic approach to explanation in cognitive and neural science has been widely advocated in recent years (Bechtel and Richardson, 1993; Machamer, Darden and Craver, 2000; Glennan, 2002; Craver, 2007; Kaplan, 2011; Piccinini and Craver, 2011; Milkowski, 2013; Boone and Piccinini, 2016b). According to this approach, many explanations in neuroscience and cognitive sciences are mechanistic. As a general formulation we can say that "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, Darden and Craver, 2000, p. 3). Mechanistic models explain phenomena by describing their underlying mechanism. For example, release of neurotransmitter when the axon is depolarized is explained by a detailed description of a process that begins with depolarization of the axon that causes opening of calcium channels, and is terminated when fusing of neurotransmitter vesicles to the cell membrane causes neurotransmitter

to be released (see Piccinini and Craver 2011 for more detail). Mechanistic explanations are hierarchical; each explanatory capacity can itself be explained mechanistically (Fig. 2). Here, the levels stand in part-whole relations. Computational and structural descriptions can exist in the same level and are both part of the same mechanistic explanation.

Fig. 2 – An example of a hierarchy of mechanistic explanations. S is the system, Ψ is the explanandum activity, X_i and P_i are the *explanans* components and Φ_i and ρ_i are their respective activities. Adapted from (Craver, 2007).



The argument that computational explanations are mechanistic is similar to the one presented in Section 2 for functional analyses; a model has not been shown to be an explanation until it has been shown to describe the actual causal structure[11] (Kaplan, 2011; Kaplan and Craver, 2011;

[11] The commitment to an 'actual causal structure' can be interpreted either ontologically or epistemically. Often, proponents of the mechanistic view maintain that explanations make ontic commitments about the causal structures in the world (Craver, 2014). Weiskopf (2011) argued that the distinction between possible models and actual explanations is epistemic and determined by the degree to which the model accommodates available evidence. This line of thought is continued in (Colombo, Hartmann and Van Iersel, 2015), who suggest an

Piccinini and Craver, 2011; Milkowski, 2013; Boone and Piccinini, 2016b). Once a model describes the actual causal structure of a phenomenon it is a *mechanistic model*. Consequently, computational models either do not describe causal relations among components and are therefore not explanatory, or they do and are therefore mechanistic. In this regard, structural information about the implementing system is taken to be necessary to confirm that a model is an explanation. Similar arguments are made for the claim that Marr's computational level is an incomplete mechanistic explanation – a mechanistic sketch (Kaplan, 2011; Piccinini and Craver, 2011; Piccinini, 2015; Boone and Piccinini, 2016b).

The mechanistic approach allows us to consider the unity of science independently from questions about multiple realization. There is only one way to explain – by describing causal structures – but higher levels need not be reducible to lower levels in the way envisioned by (Oppenheim and Putnam, 1958), which implies that all phenomena are explicable at the lowest level. Instead, phenomena at all levels are explained mechanistically and the hierarchy of mechanistic explanations connects phenomena from different levels. Further, because details about implementation are required to turn a computational model into an explanation, computational explanations cannot be taken to be autonomous from implementation, even if they are multiply realized in different implementations. By this, the mechanists answer, in a novel way, some of the challenges to the unity of science posed by the functional analysis view (Craver, 2007, chap.7; but see a criticism of this framework in Levy, 2016).

However, the mechanistic approach received many criticisms which can be divided into two main objections. The first is that computational explanations are not mechanistic and therefore an integrated-leveled framework of mechanistic explanation is incorrect. The second is that it is still unclear from the mechanists' arguments how any framework can integrate computation into mechanisms while still maintaining the explanatory value of computational description.

3.1. Arguments for the non-mechanistic nature of computational explanations

The argument that computational explanations are not mechanistic usually follows one of three lines of objection. One line of objection is that at least part of computational theory in cognitive

antirealist version of mechanistic explanation, in which coherence with existing scientific theories plays a large role in supporting mechanistic models.

neuroscience addresses certain why-questions whose answers do not track causal relations in the mechanistic sense. According to the mechanists, mechanistic explanations are causal explanations in that they track the causal structure that is relevant to the explandum phenomenon. Some argue, however, that computational explanations (models) do not necessarily refer to causal relations in answering these why-questions.

Chirimuuta (2014), for example, argues that some computational models ("interpretative models") address "the question of why nervous systems operate as they do" (Dayan and Abbott, 2001), and involve explanations which typically make reference to efficient coding principles, and not to causes. Her main example is the normalization equation that models the cross-orientation suppression of simple cell response in the primary visual cortex and in other systems. As it turned out, the response of cells in V1 is significantly reduced ("suppressed") if the preferred stimuli are super-imposed by other stimuli with different, non-preferred, orientation. Heeger (1992) accounts for the phenomenon with a normalization model that states that in addition to the excitatory input from LGN, each V1 cell also receives inhibitory inputs from its neighboring V1 cells (that are sensitive to lines in different angles). This normalization equation – which quantitatively describes the cells' responses – is later found in other parts of the nervous system (Carandini and Heeger, 2012). This, Chirimuuta says, raises the following question: "Why should so many systems exhibit behavior described by normalization equation?" The answer, she continues, is non-causal, but refers to principles of information-theory: "For many instances of neural processing individual neurons are able to transmit more information if their firing rate is suppressed by the population average firing rate" (p. 143).[12]

A second argument that computational explanations are not mechanistic is that they are not necessarily decompositional. There are network (computational) models that do not decompose the explanandum capacities into sub-components and their organization. Rathkopf (2015) argues that mechanistic explanations apply to nearly decomposable systems (Simon, 1962) where nodes in a network have more and perhaps stronger connections with each other than with nodes outside the module. Many network models, however, provide non-decompositional explanations for non-

[12] This aligns with Bechtel and Shagrir's suggestion that Marr's computational theories aim to answer certain why-questions about the relation of the computed function and the physical world, and that answering these questions does not involve causal structure. See also Rusanen & Lappi (2016) who argue that Marr's computational theories track non-causal, formal and abstract dependence relations.

decomposable systems, where part-whole decomposition is not possible. For example, a network model that accounts for patterns of traffic in a road network explains the amount of traffic in each road by appealing to dependence relations that span over the entire network. Thus explaining why a certain road connecting two edges have a lighter traffic depends on the structure and organization of the entire network; it cannot be explained by decomposing the network into separate components and their organization. Similar network-dependent properties may be identified in neural networks. A related points are made by Weiskopf (2011) who points out the existence of noncomponential models in cognitive science and by Huneman (2010) who argues that in some cases the explanation does not appeal to causal structure, but to the topological or network properties of the system[13].

A third argument is related to abstraction. According to the mechanistic framework, explanations must be situated in an actual causal structure so details about how the computational model is implemented in the physical system are required to make it an explanation. This claim is a crucial part in the argument that computational explanations are mechanistic. Levy and Bechtel (2013) argued that there are cases where the relevant explanatory information abstracts away from most structural details. Many Mechanists agree with this claim and maintain that abstract explanations are still directly constrained by implementation and hence not autonomous from it (Craver and Darden, 2013; Boone and Piccinini, 2016a; Craver, 2016; Kaplan, 2017). For example, many computational theories posit digital computation, and until it is shown that this is implemented in the structure of the system these theories are not substantiated (Piccinini and Bahar, 2013). As described in Section 2, Shapiro (2016) has pointed out that constraints do not necessarily prevent autonomy.

3.2. The problem of integration of the computational and the mechanistic

Even if one is not convinced by the previous arguments, one might wonder how computational explanations are integrated within the mechanistic hierarchy. One option is viewing computational explanations as sketches of mechanisms, and, as such, they are weak, partial or elliptical explanations. According to this picture, computational explanations nicely integrate within the mechanistic framework, in that computational properties and implementational properties belong

[13] In response, mechanists have argued that these network models turn out to be explanatory only when they describe causal structures and therefore should still be considered mechanistic (Craver, 2016).

to the same level of mechanism. When adding to the computational sketches the missing structural properties we get a full-blown mechanistic explanation of a given phenomenon.

But as Haimovici (2013) points out, this picture can prove to be problematic for the mechanistic view. Often, computational explanations are taken to be a central part of explanation in cognitive and neural sciences, even by proponents of the mechanistic view (Piccinini and Bahar, 2013). If computational explanations are merely sketches, then they cannot be considered good explanations by themselves. On the other hand, if we combine structural and computational details to get a fully-fledged mechanistic explanation, the resulting explanation is no longer medium-independent (i.e., multiply realizable in different media), which is a feature that is commonly required from computation (Piccinini and Bahar, 2013; Piccinini, 2015).

The other option is that computational explanations are medium-independent, yet can be full-blown mechanistic explanations. They are are full-blown to the extent that they refer to relevant medium-independent properties (Piccinini, 2015). But this option immediately raises two other issues. One is that computational explanations are distinct (and perhaps autonomous). Granted that computational explanations are full-blown mechanistic ones; they are nevertheless distinct from implementational mechanistic explanations. Computational explanations refer to medium-independent properties, whereas implementational explanations are about medium-dependent, implementational, properties. In other words, we can reformulate the distinctness thesis around the medium-independent/medium-dependent distinction instead of the dismissed functional/structural distinction. One might argue that the medium-independent/medium-dependent distinction suffices to support the thesis that computational explanations are distinct (and arguably autonomous) from the implementational level.

The other issue is that the relation between computational and implementational properties is often not a part-whole relationship. Piccinini proposes that:

> Computing systems, such as calculators and computers, consist of component parts (processors, memory units, input devices, and output devices), their function and organization. Those components also consist of component parts (e.g., registers and circuits), their function, and their organization. Those, in turn, consist of primitive computing components (paradigmatically, logic gates), their functions, and their

> organization. Primitive computing components can be further analyzed mechanistically but not computationally. (2015, pp. 118–119)

But, given that under the mechanistic framework explanation is decomposition into parts, one can wonder how "primitive computing components can be further analyzed mechanistically but not computationally". The logic gate is implemented as a whole by some physical, electrical, circuit. Its inputs and outputs are implemented by some physical properties, e.g., voltages. This physical circuit, including the input and output voltages, can be further analyzed mechanistically but not computationally. However, the implementation relation itself is not a part/whole relation, hence not a mechanistic analysis or explanation. For example, the physical voltages that implement the input and output channels of the gates – typically characterized by digits, 1s and 0s – are not parts of the digits. So one can provide a mechanistic explanation of the implementing voltages, but it is not clear what is the mechanistic explanation of the primitive computing components such as the implemented digits.

Furthermore, even if the logic-gate/voltage relation were a mechanistic, part/whole, relations, it is still left unclear how a higher-level computational level is analyzed mechanistically both by underlying computational and implementational levels. Take the computational level that consists of "component parts (e.g., registers and circuits), their function, and their organization". Let us call it C1. The components of C1 can be further analyzed, computationally, by the computational components of an underlying computational level C0, which "consist of primitive computing components (paradigmatically, logic gates), their functions, and their organization". However, the computational components of C1 (e.g., registers and circuits) are also realized in some implementational, medium-dependent, physical properties that belong to another mechanistic level, P. But how are P and C0 related in the mechanistic hierarchy? P and C0 must be different because the first describes physical implementation details and the latter, medium-independent properties. It is also not possible for one description to be at a lower level than the other. The properties of C0 are certainly not parts of the properties of P and vice versa. It seems that we have here two different hierarchies, one computational, C0, C1, C2, …, and one implementational, P0, P1, P2, … (presumably there are further implementational levels below P0). But it is left unclear how the two are related to each other.

Furthermore, under the possibility that computations can be multiply realized in different physical structures, the mechanist is faced with a bigger problem – not just one computational hierarchy and one mechanistic hierarchy, but many mechanistic hierarchies. The mechanist has to show how each of these can be fitted within a single computational hierarchy.

The result of the issues raised here is surprisingly similar to the framework presented in Section 2. There are two distinct explanatory hierarchies for the same system: one functional and the other structural; they do not seem to integrate. While mechanists attempt to offer an alternative to this divided view, their work is not yet complete. Those who want to argue that computational explanations are mechanistic explanations will have to address these issues.

5. Summary

This chapter focused on the relation between computation and different frameworks of levels. While the viewpoints presented here are radically different from each other, they can all be seen as positions in a debate about a central philosophical question: are computational and functional explanations distinct from neuroscientific and mechanistic explanations, or do they belong to the same kind of explanation even though they might sometimes occur at different levels? There are those who have rejected the possibility that computational explanations are inherently different from neuroscientific explanations. They favor a framework where cognitive phenomena are described in levels of part-whole relations, specifically of mechanisms, where phenomena at different levels are all explained similarly. This view allows for the same kind of explanation to persist throughout the cognitive system, at the expense of the unique explanatory power of computational explanations. The main criticism of this view is that it has not yet been shown how computational and implementational descriptions can yield a single unified explanation in a way that does not diminish the central role computation has in the explanation of cognitive and neural phenomena. .

Those who believe that functional and computational descriptions are distinct from physical descriptions must adopt a view with two levels of description, as proponents of the functional analysis view do. In contrast to the previously described view, these folks emphasize what is different between the computational and physical, and endow computational explanations with irreplaceable explanatory power and autonomy from physical descriptions. This comes at the

expense of the unity of explanation in science. Moreover, current experimental practice clearly favors an integrated approach, where physical and computational properties are commonly integrated. Marr adopts a softer stance, according to which computational explanations have a distinct explanatory role, but only when they are integrated with the algorithmic and implementation explanations is the explanation complete. His approach has received many interpretations that are consistent with both the views presented above. According to the interpretation of CL favored by us, Marr's view emphasizes the relation between the cognitive system and the environment in explanation, a relation that is often overlooked.

The questions at the heart of this debate concern fundamental issues in the philosophy of cognitive science about the relation between the computational and the physical. Surely, the ongoing debate on the correct leveled framework for computation will serve as an instructive case for the philosophy of science in general.

Acknowledgements:

We thank Jens Harbecke, Arnon Levy, Gualtiero Piccinini, Mark Sprevak and Matteo Colombo for helpful comments. This research was supported by a Grant from the GIF, the German-Israeli Foundation for Scientific Research and Development.

Bibliography

Bechtel, W. (1994) 'Levels of Description and Explanation in Cognitive Science', *Minds and Machines*, 4, pp. 1–25.

Bechtel, W. and Richardson, R. C. (1993) *Discovering Complexity: Decomposition and Localization as Strategies in Scientific Research*. Princton University Press.

Bechtel, W. and Shagrir, O. (2015) 'The Non-Redundant Contributions of Marr's Three Levels of Analysis for Explaining Information-Processing Mechanisms', *Topics in Cognitive Science*, 7, pp. 312–322.

Bermúdez, J. L. (2005) *Philosophy of Psychology: A Contemporary Introduction*. Routledge.

Block, N. (1995) 'The mind as the software of the brain', in Osherson, D. N. et al. (eds) *An Invitation to Cognitive Science*. MIT Press, pp. 170–185.

Boone, W. and Piccinini, G. (2016a) 'Mechanistic Abstraction', *Philosophy of Science*, 83, pp. 686–697.

Boone, W. and Piccinini, G. (2016b) 'The cognitive neuroscience revolution', *Synthese*, 193, pp.

1509–1534.

Buck, L. and Axel, R. (1991) ‘A Novel Multigene Family May Encode Odorant Receptors: A Molecular Basis for Odor Recognition’, *Cell*, 65, pp. 175–187.

Carandini, M. and Heeger, D. J. (2012) ‘Normalization as a canonical neural computation’, *Nature Reviews Neuroscience*, 13, pp. 51–62.

Chalmers, D. J. (2011) ‘A Computational Foundation for the Study of Cognition’, *Journal of Cognitive Science*, 12, pp. 323–357.

Chirimuuta, M. (2014) ‘Minimal models and canonical neural computations: the distinctness of computational explanation in neuroscience’, *Synthese*, 191, pp. 127–153.

Churchland, P. S. and Sejnowski, T. J. (1988) ‘Perspectives on Cognitive Neuroscience’, *Science*, 242, pp. 741–745.

Churchland, P. S. and Sejnowski, T. J. (1992) *The Computational Brain*. MIT Press.

Colombo, M., Hartmann, S. and Van Iersel, R. (2015) ‘Models, Mechanisms, and Coherence’, *British Journal for the Philosophy of Science*, 66, pp. 181–212.

Copeland, J. (1996) ‘What Is Computation?’, *Synthese*, 108, pp. 335–359.

Craver, C. F. (2007) *Explaining the Brain*. Oxford University Press.

Craver, C. F. (2014) ‘The Ontic Account of Scientific Explanation’, in Kaiser, M. I. et al. (eds) *Explanation in the Special Sciences: The Case of Biology and History*. Springer Verlag, pp. 27–52.

Craver, C. F. (2016) ‘The Explanatory Power of Network Models’, *Philosophy of Science*, 83, pp. 698–709.

Craver, C. F. and Darden, L. (2013) *In Search of Mechanisms: Discoveries across the Life Sciences*. University of Chicago Press.

Cummins, R. (1983) *The Nature of Psychological Explanation*. MIT Press.

Cummins, R. (2000) ‘“How does it work?” vs. “What are the laws?” Two conceptions of psychological explanation.’, in Keil, F. and Wilson, R. A. (eds) *Explanation and Cognition*. MIT Press, pp. 117–145.

Dayan, P. and Abbott, L. F. (2001) *Theoretical neuroscience: Computational and mathematical modeling of neural systems*. MIT Press.

Dennett, D. C. (1971) ‘Intentional Systems’, *Journal of Philosophy*, 68, pp. 87–106.

Dennett, D. C. (1994) ‘Cognitive science as reverse engineering: Several meanings of “top-down” and “bottom-up”’, in Prawitz, D., Skyrms, B., and Westerstahl, D. (eds) *Logic, metholodology and philospohy of science IX*. Elsevier Science, pp. 679–689.

Egan, F. (1995) ‘Computation and Content’, *Philosophical Review*, 104, pp. 181–203.

Egan, F. (2010) ‘Computational models: a modest role for content’, *Studies in History and Philosophy of Science*, 41, pp. 253–259.

Egan, F. (forthcoming) ‘Function-Theoretic Explanation and Neural Mechanisms’, in Kaplan, D. M. (ed.) *Integrating Psychology and Neuroscience: Prospects and Problems*. Oxford University Press.

Fodor, J. (1968) *Psychological Explanation: An Introduction To The Philosophy Of Psychology*. Random House.

Fodor, J. (1974) ‘Special Sciences (Or: The Disunity of Science as a Working Hypothesis)’, *Synthese*, 28, pp. 97–115.

Fodor, J. (1994) *The elm and the expert*. MIT Press.

Fresco, N. (2014) *Physical Computation and Cognitive Science*. Springer.

Frixione, M. (2001) ‘Tractable Competence’, *Minds and Machines*, 11, pp. 379–397.

Fyhn, M. *et al.* (2004) ‘Spatial Representation in the Entorhinal Cortex’, *Science*, 305, pp. 1258–1264.

Glennan, S. (2002) ‘Rethinking Mechanistic Explanation’, *Philosophy of Science*, 69, pp. S342–S353.

Haimovici, S. (2013) ‘A Problem for the Mechanistic Account of Computation’, *Journal of Cognitive Science*, 14, pp. 151–181.

Haugeland, J. (1978) ‘The Nature and Plausibility of Cognitivism’, *Behavioral and Brain Sciences*, 1, pp. 215–226.

Heeger, D. J. (1992) ‘Normalization of cell responses in the cat striate cortex’, *Visual Neuroscience*, 9, pp. 181–197.

Horgan, T. and Tienson, J. (1994) ‘A Nonclassical Framework for Cognitive Science’, *Synthese*, 101, pp. 305–345.

Huneman, P. (2010) ‘Topological explanations and robustness in biological sciences’, *Synthese*, 177, pp. 213–245.

Kaplan, D. M. (2011) ‘Explanation and description in computational neuroscience’, *Synthese*, 183, pp. 339–373.

Kaplan, D. M. (forthcoming) ‘Neural computation, multiple realizability, and the prospects for mechanistic explanation’, in Kaplan, D. M. (ed.) *Explanation and Integration in Mind and Brain Science*. Oxford University Press.

Kaplan, D. M. and Craver, C. F. (2011) ‘The Explanatory Force of Dynamical and Mathematical Models in Neuroscience : A Mechanistic Perspective’, *Philosophy of Science*, 78, pp. 601–627.

Lee, H., Ekanadham, C. and Ng, A. Y. (2007) ‘Sparse deep belief net model for visual area V2’, in *Advances in Neural Information Processing Systems*, pp. 873–880.

Levy, A. (2016) ‘The unity of neuroscience: a flat view’, *Synthese*, 193, pp. 3843–3863.

Levy, A. and Bechtel, W. (2013) ‘Abstraction and the Organization of Mechanisms’, *Philosophy of Science*, 80, pp. 241–261.

Loewenstein, Y. and Seung, H. S. (2006) ‘Operant matching is a generic outcome of synaptic plasticity based on the covariance between reward and neural activity’, *PNAS*, 103, pp. 15224–15229.

Lycan, W. (1987) ‘The Continuity of Levels of Nature.’, in *Conciousness*. MIT Press.

Machamer, P., Darden, L. and Craver, C. F. (2000) ‘Thinking About Mechanisms’, *Philosophy of Science*, 67, pp. 1–25.

Marr, D. (1982) *Vision*. MIT Press.

Milkowski, M. (2013) *Explaining the Computational Mind*. MIT Press.

Newell, A. (1980) ‘Physical Symbol Systems’, *Cognitive Science*, 4, pp. 135–183.

Newell, A. (1982) ‘The Knowledge Level’, *Artificial Intelligence*, 18, pp. 87–127.

Oppenheim, P. and Putnam, H. (1958) ‘Unity of Science as a Working Hypothesis’, in Feigl, H., Scriven, M., and Maxwell, G. (eds) *Minnesota Studies in the Philosophy of Science, Vol. 2*. Minnesota University Press.

Piccinini, G. (2007) ‘Computing Mechanisms’, *Philosophy of Science*, 74, pp. 501–526.

Piccinini, G. (2015) *Physical Computation: A Mechanistic Account*. Oxford University Press.

Piccinini, G. and Bahar, S. (2013) ‘Neural Computation and the Computational Theory of Cognition’, *Cognitive Science*, 34, pp. 453–488.

Piccinini, G. and Craver, C. F. (2011) ‘Integrating psychology and neuroscience: functional analyses as mechanism sketches’, *Synthese*, 183, pp. 283–311.

Polger, T. (2004) *Natural Minds*. MIT Press.

Putnam, H. (1967) ‘The Nature of Mental States’, in Capitan, W. and Merrill, D. (eds) *Art, Mind, and Religion*. University of Pittsburgh Press.

Pylyshyn, Z. W. (1980) ‘Computation and cognition: Issues in the foundation of cognitive science’, *Behavioral and Brain Sciences*, 3, pp. 111–132.

Pylyshyn, Z. W. (1984) *Computation and Cognition*. MIT Press.

Pylyshyn, Z. W. (1989) ‘Computing in cognitive science’, in Posner, M. I. (ed.) *Foundations of Cognitive Science*. MIT Press.

Ramsey, W. (2007) *Representation Reconsidered*. Cambridge University Press.

Rathkopf, C. (2015) 'Network representation and complex systems', *Synthese*, pp. 1–24.

van Rooij, I. (2008) 'The Tractable Cognition Thesis', *Cognitive Science*, 32, pp. 939–984.

Roy, D. S. *et al.* (2017) 'Distinct Neural Circuits for the Formation and Retrieval of Episodic Memories', *Cell*, 170, pp. 1000–1012.

Rusanen, A. and Lappi, O. (2007) 'The Limits of Mechanistic Explanation in Neurocognitive Sciences', in *Proceedings of the European cognitive science conference*. London: Francis and Taylor.

Rusanen, A. and Lappi, O. (2016) 'On computational explanations', *Synthese*, 193, pp. 3931–3949.

Shagrir, O. (2006) 'Why we view the brain as a computer', *Synthese*, 153, pp. 393–416.

Shagrir, O. (2010) 'Marr on Computational-Level Theories', *Philosophy of Science*, 77, pp. 477–500.

Shagrir, O. and Bechtel, W. (forthcoming) 'Marr's Computational Level and Delineating Phenomena', in Kaplan, D. M. (ed.) *Integrating Psychology and Neuroscience: Prospects and Problems*. Oxford University Press.

Shapiro, L. A. (1997) 'A Clearer Vision', *Philosophy of Science*, 64, pp. 131–153.

Shapiro, L. A. (2016) 'Mechanism or Bust ? Explanation in Psychology', *British Journal for the Philosophy of Science*, axv062.

Simon, H. A. (1962) 'The Architecture of Complexity', *Proceedings of the American Philosophical Society*, 106, pp. 467–482.

Sprevak, M. (2010) 'Computation, individuation, and the received view on representation', *Studies in History and Philosophy of Science Part A*, 41, pp. 260–270.

Sterelny, K. (1990) *The Representational Theory of Mind*. Blackwell.

Stich, S. P. (1983) *From Folk Psychology to Cognitive Science: The Case Against Belief*. MIT Press.

Weiskopf, D. A. (2011) 'Models and mechanisms in psychological explanation', *Synthese*, 183, pp. 313–338.