TOPICS
TOPICS IN COGNITIVE SCIENCE

Topics in Cognitive Science (2015) 1–13

ISSN:1756-8757 print / 1756-8765 online
DOI: 10.1111/tops.12133

Marr's Attacks: On Reductionism and Vagueness

Chris Eliasmith, Carter Kolbeck

Department of Philosophy, University of Waterloo

Received 1 July 2013; received in revised form 28 October 2013; accepted 22 November 2013

Abstract

It has been suggested that Marr took the three levels he famously identifies to be independent. In this paper, we argue that Marr's view is more nuanced. Specifically, we show that the view explicitly articulated in his work attempts to integrate the levels, and in doing so results in Marr attacking both reductionism and vagueness. The result is a perspective in which both high-level information-processing constraints and low-level implementational constraints play mutually reinforcing and constraining roles. We discuss our recent work on Spaun—currently the world's largest functional brain model—that demonstrates the positive impact of this kind of unifying integration of Marr's levels. We argue that this kind of integration avoids his concerns with both reductionism and vagueness. In short, we suggest that the methods behind Spaun can be used to satisfy Marr's explicit interest in combining high-level functional and detailed mechanistic explanations.

Keywords: Marr; Reductionism; Neural engineering framework; Spaun; Semantic pointers; Cognitive modeling

1. Introduction

Marr's approach to understanding information processing in natural systems distinguishes between three levels of abstraction. The highest level identifies what computation is to be performed and why it is to be performed. The middle level focuses on representations and algorithms that will realize the identified computation. The lowest level describes how the relevant algorithm can be implemented in a physical substrate. In the context of cognitive systems, the three levels correspond to a cognitive task, a set of representations and an algorithm required to carry out the cognitive task, and a description of how neurons and can implement the algorithm.

Many researchers who have adopted Marr's three-level distinction presume that the levels are *independent*. Pylyshyn (1984), for instance, refers to them as "three

Correspondence should be sent to Chris Eliasmith, Department of Philosophy, University of Waterloo, ON, N2L 3G1, Canada. E-mail: celiasmith@uwaterloo.ca

autonomous levels of description" (p. 259). Marr himself occasionally makes remarks that seem consistent with this view: "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" (1982, p. 19). However, Marr is often careful to qualify these kinds of claims: "Such [computational] analysis does not usurp an understanding at the other levels—of neurons or of computer programs—but it is a necessary complement to them, since without it there can be no real understanding of the function of all those neurons" (1982, p. 19).

Marr stresses that it is important to understand which of the three levels are most suited to describe a phenomena (e.g., visual after-image effects are best described at the implementation level, but the ambiguity of a Necker cube is best described at a higher level). However, he does not necessarily see even the "most suited" level to be independent of the others. After all, Marr (1982) notes that some algorithms are more easily performed by certain hardware: "The choice [of algorithm], then, may depend on the type of hardware or machinery in which the algorithm is to be embodied physically" (p. 24). This suggests that Marr promotes an approach that is sensitive to interactions between different levels of analysis.

Overall, we believe that Marr's view is more nuanced than the "independence" view sometimes ascribed to him. We suspect that Marr's view is a consequence of his being motivated by a desire to combat two trends he found disturbing in the behavioral sciences community at the time (and which, we would argue, are still present today). One trend resulted in researchers looking to reduce complex cognitive systems to their basic biological constituents to explain higher level function. A second trend resulted in researchers using vague, high-level descriptions of cognitive processes which were not empirically testable. Marr explicitly attacks what he saw as the causes of both of these trends; that is, he attacks both reductionism and vagueness.

With respect to reductionism, Marr was interested in ensuring the centrality of not only mechanisms but also of their *function* to our generation of brain theories (Marr & Poggio, 1977). He found overzealous reduction in complex systems to biological elements unsatisfactory because they did not adequately account for the evident high-level functions of those systems.

With respect to vagueness, Marr wanted to ensure that our high-level descriptions of neural phenomena could be tested against empirical data (Marr, 1975). He was explicitly unsatisfied with the suggestion that under-specified conceptual theories provided deep understanding of the mechanisms of cognition.

Indeed, Marr seems interested in the combination of functional and detailed mechanistic explanations: "The fundamental point is that in order to understand a device that performs an information-processing task, one needs many different kinds of explanations" (1982, p. 4). He goes further to claim that, more than just needing various explanations, they must be simultaneously satisfying: "One of the fascinating features of information-processing machines is that in order to understand them completely, one has to be satisfied with one's explanations at many different levels" (1982, p. 4).

Above all, it seems, Marr was striving for unified models of brain function. Rolls (2011) suggests that empirical and theoretical limitations, more than a lack of desire to perform integration, is a likely explanation for Marr's own shift from neurally detailed models to more abstract, computational ones. As Rolls has observed: "Thus a more modern approach, which is making very fast progress at present, is to combine empirical neurophysiological and neuroanatomical data with approaches that produce and test theories of how the brain computes" (2011). We believe that the behavioral sciences are now in a position to realize the kind of integration that Marr envisioned four decades ago. After discussing Marr's attacks against reductionism and vagueness in more detail, we discuss our recent work that we feel clearly demonstrates the benefits of, and unavoidable need for, a unifying integration of Marr's levels. We take it that, regardless of the outcome of rhetorical analyses of Marr's work, he would find this result deeply intellectually satisfying.

2. Against reductionism

Marr uses an analogy to convey the perils of reductionist thinking: " … trying to understand perception by studying only neurons was like trying to understand bird flight by studying only feathers: It just cannot be done" (1982, p. 27).

In both the source and target of the analogy, a clear conceptual separation of mechanism from function is clearly lacking. Feathers are part of the mechanisms that birds use to fly, and neurons are part of the mechanisms that animals use to perceive. However, a more detailed understanding of a feather or a neuron is not enough to understand the systems in which they operate. For a more complete understanding of a system, Marr notes that we must ask "why" questions (Marr, 1982). We can see how feathers respond to different aerodynamic forces, and how neurons respond to electrical inputs, but understanding of flight or perception comes when we ask why the mechanisms are the way they are. What function is it that they have been designed to bring about?

While the shortcomings of reductionism may be apparent, it is not difficult to understand the appeal it holds in some fields of the behavioral sciences, especially the neurosciences. Although we may recognize the importance of understanding information-processing systems from a computational point of view, neural mechanisms are often more directly accessible to study and quantification. By stimulating and measuring electrical activity in neurons, we can carefully characterize the computations that neurons perform. As our understanding improves, our characterizations can become more and more sophisticated. However, it is not always clear how much of that sophistication is required to explain basic functional properties of the system. One role that Marr's three levels can play is to highlight the need for explicit, quantitative tests of how low-level interactions give rise to high-level behavior; that is, an explicit need for characterizing the *relationships* between implementation, representation, and computation.

3. Against vagueness

Marr was clearly concerned that the empirically detailed study of mechanisms often resulted in the neglect of function (Marr & Poggio, 1977). However, he was also concerned that describing a high-level function of relevance to cognitive systems often resulted in the neglect of sufficiently specifying the relation of that function to its implementation to support empirical study. For Marr, tying a functional description of an information-processing task to specific algorithms and representations and then to implementational details would provide for empirical testability.

This was important for Marr because a specific algorithm may accomplish a computational task very efficiently, but that algorithm may not lend itself to implementation in neurons or may not correspond to the known functional divisions of the brain (recall his comment: “The choice [of algorithm], then, may depend on the type of hardware” cited previously). If a problem is characterized solely at a computational, or even algorithmic level, it would not be possible to investigate the plausibility of the brain performing the computation in the proposed way. Marr was concerned that this allows for vague descriptions that can be used to match only high-level behavioral observations, without an *understanding* of the mechanisms involved. When such vague theories do not match observations, they might be tweaked without the constraints of biology.

This need for a way to hold cognitive theories accountable to the biology in which they are implemented seems to have been a motivating factor for the development of Marr’s three levels. Indeed, it forces the analysis of information-processing systems to be explicitly related to biology. In his own words:

> A retinal physiologist would not accept this [pure information-processing account], because he would want to know exactly ***how*** the retina computes this term. A receptor chemist, on the other hand, would scarcely admit that these sorts of consideration have anything at all to do with the retina! Each point of view corresponds to a different level of explanation, and all must eventually be satisfied. (1982, p. 337)

While Marr occasionally expresses uncertainty at how this satisfaction will occur, he takes this to be a critical, ultimate goal. Indeed, he sometimes complains that “The complexity barrier is just too great. But we have started to do it, don’t forget!” (1982, p. 349). Consequently, it is clear that characterizations of Marr’s view as one of “autonomy” (see, e.g., Pylyshyn, 1984) is a straw man. Marr quite explicitly realizes how important it is to integrate theories across levels.

As neural measurement techniques have continued to advance over the last 40 years, we can now much more thoroughly test a high-level description if it is linked to neurons. In the next section, we briefly outline a mathematical method for forming such a link. In the section that follows, we describe the result of applying this theory to constructing what is currently the world’s largest functional brain model (Eliasmith et al., 2012).

Analysis of this case allows us to clearly characterize how Marr's levels interact in a model that, we believe, realizes his vision of integrating across levels.

4. The neural engineering framework

The neural engineering framework (NEF) can be seen as a way to characterize the relationship between Marr's highest two levels and the level of implementation (Eliasmith & Anderson, 2003). Specifically, the three principles of the NEF provide a way to (a) encode information represented in a high-level vector space into a population of low-level neurons; (b) decode computations defined over that vector space; and (c) specify dynamics using representations and computations that can be implemented in neurons. Together, these quantitative principles provide a means of implementing specific algorithms in spiking neural models by computing the needed connection weights between neural populations.

To better understand the steps involved, let us consider the example of constructing a neural model that can remember a state. The computation in this case is remembering a state, the algorithm will be to remember the state by effectively recycling the current state over time. We can express this algorithm as an equation:

$$\frac{dx(t)}{dt} = Ax + u(t) \tag{1}$$

This equation states that the change in the state x is the current state times A plus some input $u(t)$. If we set $A = 0$ with no input, the state will not change and we will have a memory. This dynamical system specifies an integrator.

To employ the NEF, we specify an encoding of the state x (of arbitrary dimensionality) into a population of neurons a. The firing rate of each neuron changes depending on what state is being encoded, as governed by this encoding equation:

$$a_i(x) = G_i\left[\alpha_i e_i x + J_i^{bias}\right] \tag{2}$$

In Eq. 2, each neuron has an associated "preferred direction vector" e of the same dimensionality as the input vector x. The similarity between the neuron's preferred direction vector and the input vector (as measured by a dot product) determines how active the neuron will be for a given input. The equation determines how much current is injected into a neuron modeled by the nonlinearity G_i for a given input, x, gain, α, and bias (or background current), J^{bias}. The current determined by Eq. 1 is used as the input to a nonlinear neuron model, which determines the spiking behavior of a neuron.

Eq. 1 allows us to determine the spiking behavior of a population of neurons given some represented input vector. To fully determine what information is carried by that representation, we can decode the spiking activity of a population of neurons to determine what vector it is encoding. Specifically, we can find decoders d for a population of neurons using Eq. 3 that can be used to scale the activities of the neural ensemble to give back an estimate of the input signal $\hat{x}$ (Eq. 4). The decoders are found using:

$$d = \Gamma^{-1}\Upsilon \tag{3}$$

where $\Gamma = \int a_i(x)a_j(x)dx$ is a correlation matrix, and $\Upsilon = \int a_i(x)a_j(x)xdx$ determines the space being optimized for. The resulting decoders, d, are least-squares optimal linear decoders, which give an estimate of the encoded space:

$$\hat{x} = \sum_i a_i(x)d_i \tag{4}$$

In essence, Eqs. 2–4 specify the first principle of the NEF: the representation principle. Having specified the representations, we can then specify the transformations of interest using the second principle: the transformation principle. In this example, the only transformation is a linear one, A. As a result, it will be incorporated directly into the weights we compute (see Eq. 5). For nonlinear transformations, we can compute a different set of optimal linear decoders d^f using a slight variation on Eq. 3 (see Eliasmith & Anderson, 2003, for details).

The third principle—the dynamics principle—determines how to map dynamics, like those in Eq. 1, into a neural system with biologically determined dynamics. Specifically, we must compute the transformation $A^{'} = \tau A + I$, where τ is the synaptic time constant of the neural connection, and I is the identity matrix. In this case, $A = 0$ so $A^{'} = I$.

We now use our application of the principles to directly compute the necessary connection weights among the neurons. In this case, there is only a single recurrent connection (in general, if $A^{'}$ is non-zero there is a recurrent connection). In the NEF, the connection between any two populations a_i and b_j is given by:

$$\omega_{ij} = \alpha_j e_j A' d_i \tag{5}$$

Because there is a recurrent connection in this memory circuit, a_i and b_j refer to the same population. Regardless, connecting the neurons with these weights will result in the dynamics specified by Eq. 1. We have shown elsewhere that this method allows the construction of nonlinear dynamical systems in high-dimensional spaces, is applicable for varying degrees of detail in the neural model (G_i), and has several other desirable properties, including efficient simulation, efficient computation of weights, and so on (Eliasmith, 2013; Eliasmith & Anderson, 2003). The memory model described above (and many others) can be downloaded and run in the open-source software package *Nengo* (http://nengo.ca).

5. Spaun

The principles of the NEF (and their embodiment in *Nengo*) have been used to create several large-scale spiking neural models. One such model, the Semantic Pointer Architecture

Unified Network (Spaun), is capable of performing eight perceptual, motor, and cognitive tasks (Eliasmith et al., 2012). For input, Spaun uses a single eye, which is able to read 28×28 pixel images, typically of digits and letters. These characters serve two purposes: to tell Spaun which task to perform, and to present the stimuli on which the task is performed. For output, Spaun has a simulated arm that it uses to write digits as its responses.

Spaun uses 2.5 million spiking neurons connected using the principles of the NEF, so as to be consistent with known neuroanatomy. The model matches known physiology of the areas included in the model, and employs 4 classes of neurotransmitter (GABA, NMDA, Dopamine, and AMPA) in a manner consistent with known physiology. The eight tasks that Spaun performs are (a) copy drawing, in which Spaun reads in a character and attempts to draw that character in the same style of handwriting; (b) image recognition, in which Spaun views a character and draws the character in its default handwriting; (c) reinforcement learning, in which Spaun performs a three-armed bandit task determining which of three possible choices it believes to be the most rewarding; (d) serial working memory, in which Spaun views a sequence of characters and attempts to reproduce it by drawing the characters in the order in which they were read; (e) counting, in which Spaun reads a number to start at and a number indicating the number of times to count up from the start number, and then writes the final value; (f) question answering, in which Spaun is given a list and asked one of two questions: what number is at a specific position of the list, or at what position is a specific number; (g) rapid variable creation, in which Spaun is asked to complete a sequence based on earlier given examples of similar sequences; and (h) fluid reasoning, in which Spaun is asked to solve a problem from the Raven's Progressive Matrices intelligence test. The model matches well to known spike patterns, neural oscillations, single-cell tuning, and a variety of behavioral measures (e.g., accuracy, speed, response variability; Eliasmith et al., 2012).

Employing Marr's three levels, we would claim that these tasks are performed by implementing algorithms underlying a wide array of computations. Specifically, there are populations of neurons covering 20 different neuroanatomical areas, including visual, motor, frontal, and subcortical areas. We have identified six major functional components that span these areas, including action selection, visual compression, motor control, working memory, and so on. Each of these components implements transformations on specific representations for that area.

However, we describe all the representations as species of "semantic pointers," a compressed neural representation of high-dimensional vector spaces (Eliasmith, 2013). Semantic pointers allow for high-dimensional information to be compressed into lower dimensional information that can be more efficiently manipulated. For example, Spaun would initially represent the image of the number that it sees as a high-dimensional vector containing the values of all 28 x 28 pixels. However, Spaun's visual system compresses this information to a 50-dimensional space that can be used to characterize its visual input in terms of higher level features and ultimately categories. Depending on the task at hand, these visual representations can be used to drive the arm (e.g., for copy drawing), or to elicit conceptual representations to do more sophisticated semantic or syntactic processing.

These representations, over which Spaun's algorithms are defined, are represented in neurons using the principles of the NEF, completing the link to implementation.

6. Spaun as integrating Marr's levels

Given this characterization, it should be clear that the behavior of Spaun can be analyzed at each of Marr's three levels. The definition of the tasks and the computations required to complete them are the highest level, the representation of information as semantic pointers and the manipulation of the semantic pointers are the middle level, and the implementation of the semantic pointers and their accompanying algorithms into neurons as described by the NEF are the lowest level. While each level can be identified independently, the *integration* of the levels is critical to the development, and understanding, of a model like Spaun.

The algorithms used to implement the computations that Spaun performs are guided by both what is known about the function of specific brain areas and the known physiological properties of those areas. For example, the basal ganglia (BG) have long been thought to be involved in action selection. Consequently, Spaun includes the hypothesis that this anatomical area acts as a general action selector using a variation on a winner-take-all approach. The detailed anatomy of this brain area greatly constrains the specific algorithms Spaun uses to do a winner-take-all-like computation with the BG—in a way that employs the physiological and anatomical properties of neurons in this area (e.g., GABA, an inhibitory neurotransmitter, is very common in the BG). Without the details of the neuroanatomy and neurophysiology of this brain region, completely different algorithms could have been used to accomplish certain tasks, but it is unlikely that a similarly wide variety of data could have been accounted for. More to the point, some of the data constraining the model are implementational and other data are behavioral—only a model constructed at both levels could hope to unify these two sources of information.

Perhaps the best example from Spaun on the importance of an integrated approach comes from the results of the working memory task. When recalling a list of numbers, humans have more success recalling items at the beginning and end of the list than items in the middle of the list. This improved recall of items at the beginning of the list is called the "primacy effect." Improved recall of items at the end of the list is called the "recency effect." The working memory model used in Spaun reproduces this result (see Fig. 1).

In constructing this working memory model, we first identified the computation (serial working memory), we then identified the algorithm (a simple convolutional algorithm defined over semantic pointers), and finally we implemented the algorithm in neurons using the NEF. Because we specify the algorithm separately, we can directly simulate the equations of the algorithm without implementing it in neurons. However, doing so does not do a good job of capturing the behavioral data (see Fig. 2). As a result, implementation of the algorithm in a neural model is critical to the good explanation of the behavioral data. The algorithmic and implementational levels are clearly interdependent.

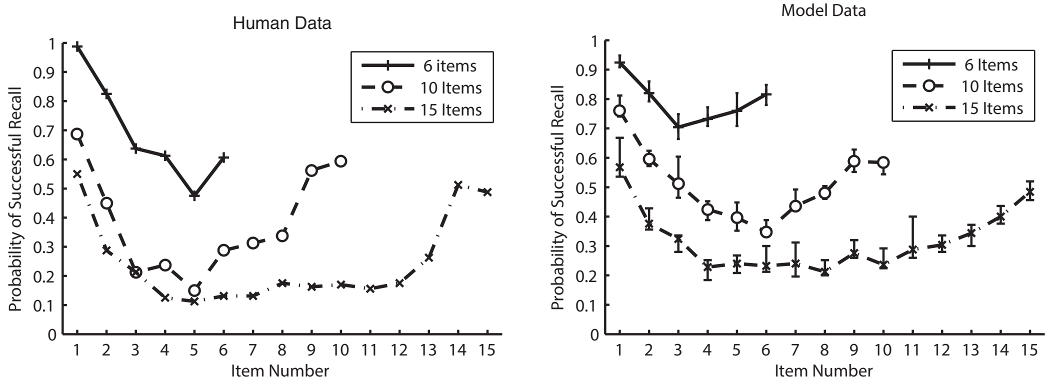

Fig. 1. Human and model data for a serial working memory task. Note the primacy and recency effects evident in both. (From Eliasmith, 2013.)

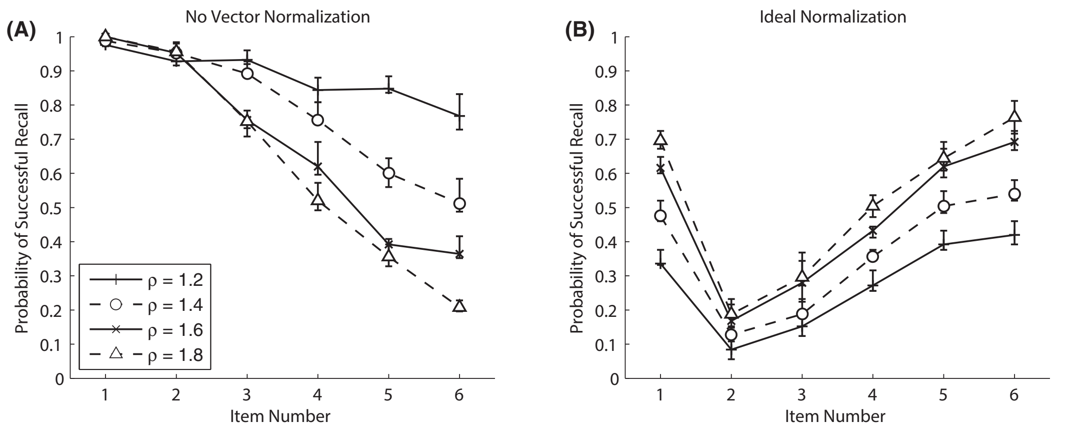

Fig. 2. The results of simulating the serial working memory equations without neurons. (A) Simulation using no vector normalization. (B) Simulation that includes vector normalization. Neither case captures the primacy and recency effects of the human data in Fig. 1 well. (Figure from Eliasmith, 2013.)

These results show the importance of thinking about multiple levels simultaneously when constructing information-processing systems. In fact, in developing the model, we ran the nonneuron model of the working memory task first and considered moving to a different, more complex algorithm. However, because implementation in neurons does not perfectly reproduce such mathematical models, we tested the neural model. To our surprise (and relief), the implemented model did an excellent job of matching the behavioral data. We now know that this is because neural saturation effects cause a kind of "soft normalization" to happen to the vector space being encoded by the memory. This results in a specific accuracy curve in the behavioral data that is not reproduced with either no normalization or with standard "perfect" normalization (see Fig. 2). The success of the model depends on how two levels of Marr's hierarchy interact. This is a lesson that echos Marr's earlier quoted observations.

Perhaps even more significantly, inclusion of this model of working memory into a large-scale model like Spaun helps to highlight problems with some past models of working memory. There are, for instance, a class of working memory models known as “slot models” that presume that working memory is implemented as a series of physically separate slots that can take different fillers (Zhang & Luck, 2008). While these models are able to explain a variety of psychological data on a few specific tasks, it is unclear how this encoding could be used for the vast majority of tasks. For example, there is no suggestion as to how order information could be included in a slot model. Since usually only a small number of slots are assumed (e.g., three or four), it is unclear how more than that many items can be encoded (in contrast, the Spaun working memory can encode arbitrary list lengths). Without order information, none of the working-memory-dependent tasks that Spaun performs could be accomplished.

The differences between Spaun and slot models also result in different predictions regarding the expected results of neural measurements. Spaun suggests a specific pattern of similarity between the neural activity during encoding of a single item in working memory, versus encoding that same item along with other items. Specifically, the similarity of neural firing in Spaun drops off exponentially as items are added. This prediction contradicts that from slot models, in which the similarity stays constant.

In short, adopting an integrated view can help to (a) relate implementational predictions (and measurements) to high-level model function (exemplified by neural-firing similarity predictions); (b) ensure that a specific model retains its utility when we adopt a fuller perspective of brain function (exemplified by the consequence of not including order information in a working memory mode); and (c) determine which effects may be appropriately captured with high-level models and which from implementational details (exemplified by the neural implementation of a high-level, working memory model capturing primacy and recency effects).

7. Spaun and reductionism

Reductionism is alive and well in the neural modeling. Indeed, the massively funded €1 billion Human Brain Project (HBP) is concerned primarily with developing very detailed computational models of individual neurons. The hope is that connecting millions or billions of these neurons together will lead to insights about how the brain works. Unfortunately, the number of ways such neurons could be connected are nearly limitless. Adopting the approach of matching the low-level connection statistics is unlikely to result in recognizable function if, as the NEF suggests, connections depend on the specific properties of each cell (i.e., the encoder, gain, and decoder). If such a model cannot be related to functions like remembering, seeing, moving, and so on, the dream of understanding the neural basis of disease is unlikely to be realized. These same kinds of concerns were voiced by Marr four decades ago. Implementation without function, he argued, will not lead to understanding.

Spaun, in contrast, demonstrates how both implementation and function can combine to analyze and develop mechanistic information-processing models. The many areas included in the model have specific functions, and all are implemented in a spiking neuron substrate. Granted, the neurons Spaun employs (leaky integrate-and-fire neurons) are simpler than those used in the HBP. However, the NEF is defined over generic neural nonlinearities, and it is not always obvious that more low-level detail makes for a better explanation. If there are specific low-level questions Spaun is not able to answer, increased detail can be introduced into the neural models in the relevant areas. We may be concerned that additional detail will "break" a given model. We suspect that this will depend on precisely how specific details were "simplified away." In many cases, such simplifications will not matter[1] but in other cases they could well change the effects of considering implementational details.

Regardless, it seems that in short, there is no one, correct level of detail for all models (Eliasmith & Trujillo, 2013). The required level of explanation is bound to be sensitive to the questions being asked and the explanations being demanded. We believe that the flexibility the NEF provides for incorporating details as needed results in a practical tool for generating models at the *relevant* (not *right*) level of detail. Critically, the NEF also relates the implementational and algorithmic levels—consistent with Marr's view that many, simultaneously accurate explanatory levels are required to understand information-processing systems.

8. Spaun and vagueness

We believe one of the most compelling aspects of the Spaun model is that it matches data at many scales of analysis and from many different disciplines. Although researchers often worry that a large-scale model like Spaun has many parameters, and hence is easy to fit to data, the opposite is true. Because Spaun can be described at each of Marr's three levels, and because its assumptions can be tested using highly disparate data sets (e.g., those from molecular neuroscience, systems neuroscience, psychophysics, psychology, etc.), it turns out to be a highly constrained model. This serves to eliminate vagueness not only at a single level, but across all three levels identified by Marr. In short, any behavior that Spaun performs can be checked against multiple types of human and animal data.

As a result, Spaun not only succeeds at tasks in a manner similar to humans, but it also fails at tasks in a similar manner (as shown in the results of the serial working memory task in the earlier section). As discussed with respect to the serial working memory component, the specific profile of failure, which was never considered at any of the levels individually, results from the integration of all three levels. Furthermore, the model makes specific testable predictions regarding several aspects of working memory. For instance, it predicts the kinds of single neuron responses that are expected during both successful and unsuccessful recall in serial working memory (a kind of data that are largely lacking, given the dearth of serial working memory experiments on animals). As detailed in the original Spaun paper, this serial working memory model makes specific predictions regarding the expected change in similarity of neural activity as items are added to memory, which can distinguish slot-based hypotheses from those employed by Spaun (Eliasmith et al., 2012).

These kinds of specific predictions, coupled with the many sources of data that can be drawn on to quantitatively constrain the model, directly address Marr's concern with the vagueness of purely functional models.

9. Conclusion

We believe that Marr, though often interpreted otherwise, held a deep conviction that the behavioral sciences ought to generate unified, integrated models of information processing in cognitive systems across levels of analysis. Marr's own career spanned detailed neurophysiological models, and high-level functional models. Our work has been directly focussed on developing methods for not only spanning, but also integrating models across Marr's levels. Currently, Spaun provides the most comprehensive attempt we have made at doing so. Given his deep understanding of brain function, there are no doubt many problems that Marr would find with the Spaun model. Nonetheless, we believe that he would still appreciate the attempt.

We also believe the methods we have employed directly support the complementary, critical roles of both low-level modeling and high-level, cognitive-processing analysis. The working memory model that we discussed previously would not have been generated by merely focussing our attention on the tuning properties of individual neurons. If we did not have the high-level mathematical model available, it would not have been evident how to use the NEF at all. The NEF, after all, does not tell us how the brain works. Rather, if we have a (high-level, cognitive) hypothesis about brain function, the NEF can be used to determine if it can be plausibly implemented in a spiking neural network that incorporates the implementational constraints of a particular brain area. Clearly, the high-level hypothesis plays a central role.

As a result, much of our recent work has been focussed on stating a coherent *set* of high-level hypotheses that are naturally implemented in the NEF. The result of this work is called the semantic pointer architecture (SPA) and is described elsewhere (Eliasmith, 2013). Spaun is one example of using this general architecture for building large-scale models.

One consequence of identifying such a set of hypotheses is that classes of algorithms become more or less *plausible*: where "plausibility" is a function of both the ability of an algorithm to support specific information processing, as well as its ability to be implemented in a biological substrate. This connection to both cognitive function and biological implementation distinguishes this approach from those that focus on one or the other.

For example, it has been suggested that the Bayesian approach is overly focused on high-level function (Jones & Love, 2011). In our recent work, we have shown how the SPA relates directly to specific Bayesian algorithms (Eliasmith, 2013, chap. 7). Identifying such a relationship allows us to significantly constrain which specific subclass of Bayesian algorithms are likely to be consistent with biological mechanisms and how they can be integrated into a fuller process model. This provides one more way to help integrate Bayesian models with mechanistic neural and psychological models, complementing those methods suggested by Jones and Love (2011).

Note

1. For example, in the working memory model, the main effect of adding neurons is introducing a saturation nonlinearity for each neuron.

References

Eliasmith, C. (2013). *How to build a brain: A neural architecture for biological cognition*. New York: Oxford University Press.

Eliasmith, C., & Anderson, C. H. (2003). *Neural engineering: Computation, representation and dynamics in neurobiological systems*. Cambridge, MA: MIT Press.

Eliasmith, C., Stewart, T. C., Choo, X., Bekolay, T., DeWolf, T., Tang, Y., & Rasmussen, D. (2012). A large-scale model of the functioning brain. *Science*, *338*, 1202–1205. doi:10.1126/science.1225266.

Eliasmith, C., & Trujillo, O. (2013). The use and abuse of large-scale brain models. *Current Opinion in Neurobiology*, *25*, doi:10.1016/j.conb.2013.09.009.

Jones, M., & Love, B. C. (2011). Bayesian fundamentalism or enlightenment? On the explanatory status and theoretical contributions of Bayesian models of cognition. *Behavioral and Brain Sciences*, *34*(4), 169–188.

Marr, D. (1975). Approaches to biological information processing. *Science*, *190*, 875–876.

Marr, D. (1982). *Vision: A computational investigation into the human representation and processing of visual information*. New York: WH Freeman.

Marr, D., & Poggio, T. (1977). From understanding computation to understanding neural circuitry. *Neurosciences Research Program Bulletin*, *15*, 470–488.

Pylyshyn, Z. (1984). *Computation and cognition: Toward a foundation for cognitive science*. Cambridge, MA: MIT Press.

Rolls, E. T. (2011). David Marr's vision: Floreat computational neuroscience. *Brain*, *134*(3), 913–916. doi:10.1093/brain/awr013W.

Zhang, W., & Luck, S. J. (2008). Discrete fixed-resolution representations in visual working memory. *Nature*, *453*, 233.